

UNIVERSIDADE FEDERAL DE SÃO CARLOS  
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM  
CIÊNCIA DA COMPUTAÇÃO

**“Uso de conhecimento de senso comum na  
educação para geração de analogias e  
similaridades”**

MURIEL DE SOUZA GODOI

São Carlos  
Outubro/2007

**Ficha catalográfica elaborada pelo DePT da  
Biblioteca Comunitária da UFSCar**

G588tm

Godoi, Muriel de Souza.

Uso de conhecimento de senso comum na educação para geração de analogias e similaridades / Muriel de Souza Godoi. -- São Carlos : UFSCar, 2008.  
86 f.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2007.

1. Senso comum. 2. Materiais instrucionais. 3. Ensino a distância. 4. Geração de analogias. 5. Geração de similaridades. I. Título.

CDD: 004.6 (20<sup>a</sup>)

**Universidade Federal de São Carlos**  
**Centro de Ciências Exatas e de Tecnologia**  
**Programa de Pós-Graduação em Ciência da Computação**

***“Uso de conhecimento de senso comum na  
educação para geração de analogias e  
similaridades”***

**MURIEL DE SOUZA GODOI**

**Dissertação de Mestrado apresentada ao  
Programa de Pós-Graduação em Ciência da  
Computação da Universidade Federal de São  
Carlos, como parte dos requisitos para a  
obtenção do título de Mestre em Ciência da  
Computação.**

**Membros da Banca:**



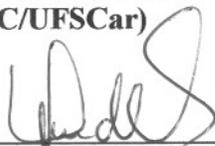
---

**Profa. Dra. Júnia Coutinho Anacleto**  
**(DC/UFSCar)**



---

**Profa. Dra. Lúcia Helena Machado Rino**  
**(DC/UFSCar)**



---

**Prof. Dr. Clóvis Torres Fernandes**  
**(ITA – São José dos Campos)**

**São Carlos**  
**Outubro/2007**

*“Sábio é o homem que estuda como se fosse viver eternamente, e vive como se fosse morrer amanhã”.*

*Autor desconhecido*

## AGRADECIMENTOS

---

A Deus, pois sem sua divina bondade nada seria possível.

Aos meus pais Alair e Ivone, que com muito amor e dedicação, ensinaram-me a base para a vida. Sempre me incentivando a continuar quando pensava em desistir.

À minha noiva Marcia, por todo o amor, carinho, e companheirismo. Além de toda a compreensão e apoio principalmente na fase final deste trabalho.

Aos meus irmãos Aline, Marcelo e Ariane, e minha avó Cida que, mesmo à distância, torceram por mim.

Aos meus grandes amigos Frederico Dutra Resende e André Andreatta, pela amizade que permanece mesmo à distância.

À orientadora e amiga Junia Coutinho Anacleto, por acreditar no meu potencial, pelas sábias orientações nas horas certas e por toda assistência fornecida durante o mestrado.

Aos meus colegas do LIA, Fabiano, Marie, Vânia, Américo, Alexandre, Alessandro, Eliane e a todos os outros, que compartilharam as alegrias e dificuldades no dia-a-dia do laboratório, aprendi muito com cada um de vocês.

Aos colegas de mestrado, pelo companheirismo durante diversas noites na sala de estudo durante o período das provas.

Ao Prof. Sérgio Roberto Pereira da Silva, orientador da minha graduação, por ter me apresentado ao fascinante mundo da pesquisa.

Aos vizinhos da kitnet, Vitor, Átila, Jeison, Rodolfo, Gabriel e Aníbal, pelas horas de descontração.

À Verinha, pelo carinho ao preparar o cafezinho que me manteve acordado no laboratório.

À CAPES, por propiciar esta pesquisa, cujos recursos possibilitaram dedicação exclusiva.

O uso de analogias é uma estratégia cognitiva frequentemente utilizada pelos alunos quando eles querem aprender algum novo conceito. Neste caso, a analogia relaciona o novo conhecimento ao conhecimento prévio do aluno, fixando-o em sua estrutura cognitiva. Pesquisas mostram que a utilização de analogias e a consideração do conhecimento prévio do aluno no material instrucional aumentam a qualidade do material gerado, estimulando os alunos a utilizarem suas estratégias cognitivas. Porém, não se encontra na literatura como a utilização de analogias com o uso do conhecimento prévio do aluno pode ser viabilizada para facilitar a educação continuada, principalmente considerando o apoio computacional para o *e-learning*. Assim, neste trabalho é apresentado um protótipo de um assistente on-line para suporte ao treinamento sobre segurança no trabalho. O protótipo usa a base de conhecimento de senso comum, e os recursos computacionais do projeto *Open Mind Common Sense* no Brasil juntamente com uma base do conhecimento sobre segurança no trabalho, para explicar termos que podem ser não familiares para os alunos. É explorado o uso de analogias para explicar os tópicos, e um método de captura de contexto são utilizados para ajudar o estudante a associar os tópicos relacionados. Acredita-se que o senso comum pode considerar aspectos culturais durante a criação de analogias e ajudando assim os estudantes. Explora-se, também, a sugestão de analogias criadas com o conhecimento de senso comum para o professor durante o desenvolvimento do material instrucional. Os resultados obtidos em um experimento realizado com estudantes apontam que eles consideraram as analogias úteis no processo de aprendizagem e que o sistema é útil no entendimento de novos conceitos, ajudando a conectar o novo conhecimento ao conhecimento de senso comum.

## ABSTRACT

---

The use of analogies is a cognitive strategy frequently adopted by learners when they want to understand some new concept. In that case, the analogy connects the new concept with the previous knowledge, attaching it in the learner cognitive structure. Some studies points out the use of analogies and previous knowledge of learners increases the learning material quality, stimulating the learners to use their cognitive strategies. Nevertheless, there are no studies about a way to enable both the use of analogies and learners previous knowledge to facilitate continued learning, mainly taking into account computational support for e-learning. In this work it is presented a prototype of an on-line assistant to support a training course about workspace safety issues. The application uses a common sense reasoning engine and the Brazilian Open Mind common sense knowledge base, to make inferences about concepts that might be unfamiliar to the students. It is explored the use of analogies to explain topics, enhancing learning by using context capture to help students associate related topics. It is believed that common sense can be used to take into account cultural considerations while helping learners to build analogies. It is also explored the suggestion of analogies, which were build using commonsense knowledge, to the teachers during the project of instructional material design. Preliminary results with students using the assistant point that they considered analogies useful in the learning process, that the system is helpful in understanding new concepts, and that it helps connect the information searched for with common sense knowledge.

## LISTA DE SIGLAS E ACRÔNIMOS

---

AE – Aprendizagem Eletrônica

API – *Application Program Interface*

EAD – Ensino à Distância

IA – Inteligência Artificial

IHC – Interação Humano-Computador

IHH – Interação Humano-Humano

JSP – *Java Server Pages*

LIA – Laboratório de Interação Avançada

LP – Linguagem de Padrões

MediaLab – *Media Laboratory*

MIT – *Massachusetts Institute of Technology*

MME – Mecanismo de Mapeamento de Estrutura

OMCS – *Open Mind Common Sense*

OMCS-Br – *Open Mind Common Sense no Brasil*

TME – Teoria de Mapeamento de Estrutura

UFSCar – Universidade Federal de São Carlos

<b>CAPÍTULO 1 - INTRODUÇÃO .....</b>	<b>1</b>
1.1. Motivação .....	1
1.2. Objetivos.....	2
1.3. Organização do Trabalho.....	2
<b>CAPÍTULO 2 - PROVENDO SENSO COMUM AOS COMPUTADORES.....</b>	<b>4</b>
2.1. Considerações Iniciais .....	4
2.2. O Conceito de Senso Comum.....	5
2.3. Bases de Conhecimento de Senso Comum .....	6
2.4. O Projeto Open Mind Common Sense no Brasil.....	13
2.4.1. A arquitetura do projeto OMCS-Br .....	13
2.4.2. A Coleta dos Dados .....	15
2.4.3. A Construção da <i>ConceptNetBr</i> .....	22
2.4.3.1. <i>Extração</i> .....	23
2.4.3.2. <i>Normalização</i> .....	24
2.4.3.3. <i>Relaxamento</i> .....	28
2.4.4. Utilização da <i>ConceptNetBr</i> .....	29
2.5. Considerações Finais .....	32
<b>CAPÍTULO 3 - ANALOGIAS GERADAS COM O CONHECIMENTO DE SENSO COMUM E O PROCESSO DE APRENDIZAGEM .....</b>	<b>34</b>
3.1. Considerações iniciais .....	34
3.2. O Conceito de Analogia .....	34
3.3. Potencialidade do Uso de Analogias com Senso Comum no Processo de Aprendizagem.....	37
3.4. Considerações Finais .....	41
<b>CAPÍTULO 4 - GERANDO ANALOGIAS ATRAVÉS DA BASE DE CONHECIMENTO DO PROJETO OMCS-BR .....</b>	<b>43</b>
4.1. Considerações Iniciais .....	43

4.2. A Teoria de Mapeamento de Estruturas - TME .....	43
4.3. O Mecanismo de Mapeamento de Estruturas - MME .....	44
4.3.1. Construção dos Mapeamentos Locais .....	45
4.3.2. Construção dos mapeamentos globais.....	46
4.3.3. Avaliação dos Mapeamentos Globais.....	49
4.4. Implementação de um algoritmo baseado no MME considerando a base OMCS-Br.....	49
4.4.1. Construção dos Mapeamentos Locais .....	50
4.4.2. Construção dos Mapeamentos Globais .....	52
4.4.3. Avaliação dos Mapeamentos Globais.....	53
4.5. Considerações Finais .....	54
<b>CAPÍTULO 5 - APLICANDO A GERAÇÃO DE ANALOGIAS EM APLICAÇÕES EDUCACIONAIS.....</b>	<b>56</b>
5.1. Considerações iniciais .....	56
5.2. Um Sistema de Apoio para Treinamento a Distância.....	57
5.2.1. A Implementação do Sistema de Apoio .....	57
5.2.2. Avaliação do Sistema Desenvolvido .....	61
5.3. Auxiliando o Professor na Criação de Analogias para o Material Instrucional .....	65
5.3.1. Cognitor.....	65
5.3.2. O padrão Correlação.....	67
5.3.3. Implementação do Assistente Para Geração de Analogias.....	68
5.3.4. Análise dos Estudos de Casos Reais Simulados no Segundo Protótipo.....	71
5.4. Considerações finais .....	75
<b>CAPÍTULO 6 - CONCLUSÕES .....</b>	<b>77</b>
6.1. Conclusões.....	77
6.2. Análise Crítica .....	77
6.3. Dificuldades a serem superadas.....	78
6.4. Publicações obtidas .....	78

6.5. Trabalhos futuros.....	81
-----------------------------	----

## LISTA DE FIGURAS

---

Figura 1. Apresentação de um gabarito do site do Projeto OMCS. ....	9
Figura 2. Diagrama da arquitetura do projeto OMCS-Br.....	14
Figura 3. Exemplo de gabarito utilizado para coletar dados no site do projeto OMCS-Br.....	15
Figura 4. Exemplo de retroalimentação para geração de novas templates.....	16
Figura 5. Gabarito exibindo cinco campos para facilitar o trabalho dos colaboradores. ....	18
Figura 6. Sistema de revisão dos fatos coletados. ....	19
Figura 7. Distribuição do número de fatos coletados por mês. ....	22
Figura 8. Exemplo do arquivo de dados utilizado pela <i>ConceptNetBr</i> . ....	23
Figura 9. Fase de extração durante a geração da <i>ConceptNetBr</i> . ....	24
FIGURA 10. Formato de saída do analisador sintático Curupira.....	26
FIGURA 11. Processo de normalização dos argumentos das relações. ....	27
Figura 12. Levando conhecimento do tipo <i>PropertyOf</i> dos nós filhos para nós pais.....	29
Figura 13. Exemplo de similaridade literal na comparação entre domínios. ....	36
Figura 14. Exemplo de analogia na comparação entre domínios.....	36
Figura 15. Exemplo de comparação entre domínios classificada como abstração.....	37
FIGURA 16. Alteração da estrutura cognitiva do aluno pela ocorrência da aprendizagem significativa. ....	39
Figura 17. Exemplificando a conformidade do uso de analogias e o processo de Aprendizagem Significativa. ....	40
Figura 18. A arquitetura do sistema de apoio para treinamento a distância.....	58
Figura 19. <i>Screenshot</i> do sistema de apoio a treinamento à distância.....	60
Figura 20. Área de conceitos relacionados.....	61
Figura 21. <i>Screenshot</i> da tela apresentada aos voluntários antes do experimento. ....	61
Figura 22. Questionário utilizado para avaliação do protótipo desenvolvido. ....	62
Figura 23. Interface principal da ferramenta Cognitor. ....	67
Figura 24 - Arquitetura do protótipo do assistente que sugere analogias ao professor.....	69
Figura 25. Interface do primeiro passo do assistente para criação de analogias. ....	69
Figura 26. Interface do segundo passo do assistente para criação de analogias.....	70
Figura 27. Professor inserindo dados sobre camisinha no primeiro passo do assistente.....	71
Figura 28. Segundo passo do assistente exibindo analogias para o termo "camisinha". ....	72
Figura 29. Professor inserindo dados sobre aparelho de pressão no primeiro passo do	

assistente.....	73
Figura 30. Segundo passo do assistente exibindo analogias para o termo "aparelho de pressão".....	74
Figura 31. Segundo passo do assistente exibindo analogias para o termo "aparelho de pressão".....	75

## LISTA DE TABELAS

---

Tabela 1. Tipos de relações semânticas da ConceptNet. Adaptada de (Singh, 2002b).....	10
Tabela 2. Conjunto de atividades e gabaritos disponibilizados no sítio OMCS-Br. ....	17
Tabela 3. Distribuição dos colaboradores por escolaridade. ....	21
Tabela 4. Distribuição dos colaboradores por faixa etária. ....	21
Tabela 5. Efeitos da normalização na rede semântica gerada. ....	27
Tabela 6. Opinião dos voluntários sobre a facilidade de uso do sistema. ....	63
Tabela 7. Opinião dos voluntários sobre a utilidade do sistema em esclarecer dúvidas. ....	63
Tabela 8. Opinião dos voluntários sobre a utilidade da analogia para entender o termo pesquisado. ....	64

# CAPÍTULO 1 - INTRODUÇÃO

---

## 1.1. Motivação

O mundo atual está organizado de forma que o conhecimento que as pessoas dominam é algo muito importante e que pode definir o seu sucesso profissional. Também a vida moderna, com seu ritmo ágil e repleta de tarefas e compromissos para serem feitos ao mesmo tempo é algo típico nos nossos dias.

Nesse contexto, a utilização dos avanços na área computacional para prover apoio ao processo continuado de aprendizagem e atribuir a ele um caráter flexível e portátil, em abordagens que combinam Aprendizagem Eletrônica (AE) e Educação à Distância (EAD) tem se tornado cada vez mais comum (Zhang *et al.*, 2004).

A utilização do conhecimento de senso comum para facilitar o processo de aprendizagem é muito importante para o seu sucesso, como constata Paulo Freire, quando estuda a questão da contextualização da aprendizagem à cultura do aprendiz (FREIRE, 1996). Porém, não se encontra na literatura como a utilização desse tipo de conhecimento de fato pode ser viabilizada para facilitar a educação continuada, principalmente considerando o apoio computacional para o *e-learning*.

Atualmente vêm-se projetos para a construção de bases de conhecimento de senso comum, buscando fazer os computadores inteligentes e capazes de prover uma interação mais útil e agradável aos seus usuários, que utilizam tecnologias computacionais para coletar e utilizar esse conhecimento. Entre eles estão o Cyc (Lenat *et al.*, 1990), o ThoughtTreasure (Mueller, 1998) e o OMCS (Singh, 2002a).

A possibilidade de utilizar a arquitetura do projeto OMCS-Br, versão brasileira do projeto OMCS, para coletar e disponibilizar conhecimento de senso comum aos brasileiros foi outra grande motivação que levou à realização deste trabalho.

Por fim, a existência de trabalhos em desenvolvimento no LIA que utilizam tal

abordagem e a possibilidade de utilizar conhecimento de senso comum para apoiar a educação continuada em ambientes de aprendizagem distintos da sala de aula foram fatores decisivos para desenvolver esta pesquisa.

## **1.2. Objetivos**

Este trabalho tem como principal objetivo demonstrar a possibilidade de utilizar conhecimento de senso comum, armazenado em uma base que está sendo construída colaborativamente por voluntários da Web, para a geração de analogias cujo objetivo é explicar conhecimento especialista, no contexto de educação a distância. Assim, o aprendiz pode, através do uso de analogias relacionadas à sua cultura e contexto, ou seja, ao seu conhecimento de senso comum, compreender mais facilmente conceitos de conhecimento especializado que sejam análogos de alguma forma com coisas que já são de seu conhecimento ou fazem parte de seu dia-a-dia.

O trabalho discute o uso automático de conhecimento de senso comum para apoiar um curso de EAD em uma companhia com mais de quinhentos mil funcionários que devem ser orientados continuamente com relação aos procedimentos de segurança no trabalho, tendo como objetivo a prevenção de acidentes no ambiente de trabalho.

Duas aplicações computacionais foram desenvolvidas usando o raciocínio baseado em senso comum considerando a base de conhecimento do projeto *Open Mind Common Sense* no Brasil – OMCS-Br: uma implementação de um assistente online que com o objetivo de fazer inferências sobre conceitos formais sobre segurança no trabalho nos quais os funcionários têm dúvidas e gerar analogias usando o conhecimento não especializado; e um assistente, que foi acoplado ao editor de material instrucional Cognitor, que sugere analogias para serem inseridas no material baseado em informações fornecidas pelo professor.

## **1.3. Organização do Trabalho**

Este trabalho está organizado da seguinte maneira: o Capítulo 2 apresenta o conceito

de senso comum e alguns projetos que visam prover esse tipo de conhecimento aos computadores, dentre eles o projeto OMCS-Br, contexto no qual se encontra este trabalho; no Capítulo 3 são apresentados os conceitos de analogia e similaridade e a relação existente entre analogia, senso comum e o processo de ensino e aprendizagem; no Capítulo 4 é apresentado o algoritmo desenvolvido neste trabalho para a geração de analogias, bem como o algoritmo utilizado como base para sua implementação; o Capítulo 5 apresenta e discute um protótipo desenvolvido para validar a abordagem proposta neste trabalho, e adicionalmente é apresentado um outro protótipo, que compartilha da mesma abordagem, porém no contexto de um editor de material instrucional; por fim, o Capítulo 6 traz algumas conclusões, disponibiliza-se uma lista de publicações relacionadas ao tema do trabalho, das quais o pesquisador participou como autor e apresenta trabalhos futuros relacionados.

## CAPÍTULO 2 - PROVENDO SENSO COMUM AOS COMPUTADORES

---

### 2.1. Considerações Iniciais

Desde o surgimento das pesquisas em Inteligência Artificial (IA), na década de 1950, os pesquisadores tentam dotar os computadores da mesma capacidade de raciocínio que o ser humano tem. Muitos avanços fizeram com que o computador ultrapassasse a capacidade dos humanos em algumas áreas, como jogar xadrez e desenhar circuitos eletrônicos. No entanto, em outras áreas o computador não é capaz de superar uma criança de cinco anos de idade, como por exemplo, ler uma história e ser capaz de responder perguntas sobre ela, ou olhar uma figura e descrever o que há nela.

Uma das linhas de pesquisa da IA, que teve como precursores Minsky (1988) e John McCarthy (1959), defende que a causa disso é que os computadores não sabem nada sobre os humanos, e os computadores precisam do conhecimento de senso comum que o ser humano tem.

Este capítulo apresenta alguns projetos que vêm sendo desenvolvidos por pesquisadores para prover o conhecimento de senso comum aos computadores, dentre eles o Projeto *Open Mind Common Sense no Brasil*, o qual foi responsável por prover os recursos computacionais que possibilitam a geração de analogias, baseada no conhecimento de senso comum dos alunos, desenvolvida neste trabalho.

Para isso, a Seção 2.2 faz uma breve discussão sobre a definição do que é o senso comum. A Seção 2.3, apresenta três projetos em que se tentam construir bases de senso comum: *Cyc*, *Thought Treasure* e *Open Mind Common Sense*. A variante brasileira do projeto *Open Mind Common Sense*, contexto no qual está inserido este trabalho, é apresentada na Seção 2.4, onde será apresentada toda a arquitetura do projeto. Finalizando este capítulo, a Seção 2.5 apresenta algumas considerações finais.

## 2.2. O Conceito de Senso Comum

Percebe-se na literatura que há diferentes definições para o conhecimento de senso comum. É um conceito de certa forma polêmico, pois é dependente do tempo, da cultura e do espaço, integrando diferentes tipos de conhecimentos. Pode-se exemplificar a explicação com duas situações do cotidiano:

- Quando as pessoas compram um filtro de barro novo, elas podem colocar algumas folhas de laranjeiras dentro do filtro para tirar o gosto de barro da água, sem saber, no entanto, quais os elementos presentes nas folhas e seus efeitos quando colocadas dentro do filtro;
- No dia-a-dia, para atravessar uma rua não se precisa saber a distância e a velocidade exata do carro e fazer as contas numa calculadora para saber se dará tempo para atravessar a rua ou não.

Esses exemplos mostram que o senso comum é um tipo de conhecimento que se acumula no cotidiano. Os seres humanos precisam do senso comum, que os permitem ter uma visão da realidade, que vem desde o hábito de fazer alguma atividade até a tradição que é passada de geração para geração.

Minsky (1988), um dos pesquisadores pioneiros na área, define senso comum como sendo “*as habilidades mentais que as pessoas compartilham*”. Ele cita ainda, que o raciocínio de senso comum é mais complexo do que o raciocínio utilizado em outras atividades específicas que parecem atrair mais a atenção e respeito do ser humano, como um jogo de xadrez. Para Minsky, essas atividades específicas envolvem muito conhecimento, mas poucos tipos de representações, ao contrário do raciocínio de senso comum que envolve muitos tipos de conhecimentos e, portanto, querer uma gama maior de habilidades.

No dicionário Houaiss da Língua Portuguesa(2001), encontra-se a seguinte definição de senso comum no contexto da filosofia romana e posteriormente no pensamento moderno:

“Conjunto de opiniões, idéias e concepções que, prevalecendo em um determinado contexto social, se impõem como naturais e necessárias, não evocando reflexões ou questionamentos; consenso.”

A questão da dependência cultural do senso comum também é citada por Mueller (1998), que considera como senso comum o conjunto de conhecimentos temporais, espaciais e de estados mentais, que incluem crenças, metas, planejamentos e emoções.

No contexto deste trabalho, considera-se o conhecimento de senso comum como sendo o conhecimento compartilhado por um grupo de pessoas levando em consideração sua cultura, o lugar e a época onde vivem. Não corresponde, portanto, necessariamente à verdade científica, pois pode incluir crenças e costumes.

### **2.3. Bases de Conhecimento de Senso Comum**

São diversos os desafios encontrados na construção de uma base de conhecimento de senso comum, considerando a quantidade e diversidade de informações necessárias para prover senso comum aos computadores.

Poder-se-ia pensar em construir um robô que vivesse no mundo e aprendesse como uma criança real, como a *Baby Machine* proposta por Turing em (1950), mas até hoje não se conseguiu construir programas que tivessem a mesma capacidade de uma criança de aprender e memorizar uma grande gama de coisas diferentes. Outra estratégia poderia utilizar técnicas de compreensão de linguagem natural para, através de textos contendo conhecimento de senso comum, o computador construir uma representação estruturada desse conhecimento. Porém, segundo Panton *et al.*(2006), isso não seria possível como uma primeira abordagem, pois, para interpretar corretamente as sentenças em linguagem natural, os computadores precisariam de conhecimento de senso comum.

Esse problema sobre o aprendizado de máquina também já foi citado décadas atrás por

McCarthy (1959). Segundo ele, para um programa ser capaz de aprender algo, primeiramente ele deve ser capaz de representar, armazenar e manipular esse conhecimento.

Outra abordagem baseia-se na idéia de que é necessário adquirir alguma experiência construindo manualmente bases de dados e programas com senso comum, para então construir máquinas que possam automaticamente aprender senso comum (Singh, 2002a; Lenat, 1995).

A maior tentativa de construir um banco de dados, de senso comum manualmente foi feita no projeto Cyc, liderado por Lenat (1995), com um grupo de engenheiros do conhecimento, que desde 1984, vêm construindo manualmente um banco de dados ao custo de dezenas de milhões de dólares e conta com 4,7 milhões de unidades de conhecimento de senso comum (Taylor *et al.*, 2007). A entrada é realizada usando CycL, uma linguagem que possibilita armazenar o conhecimento de uma forma clara e que minimiza ambigüidades.

Novos esforços têm sido feitos dentro do projeto Cyc, em que se adota uma abordagem que usa o site de busca Google para semi-automatizar a alimentação da base de senso comum (Matuszek *et al.*, 2006). Descrevendo de uma maneira sucinta, primeiramente o sistema gera temas de interesse e os transformam em strings de busca e utilizando a *API* de busca do Google, o sistema seleciona páginas relevantes. Então os dados são convertidos para a linguagem CycL com o auxílio de um analisador sintático. Em seguida, utilizando a base já existente e novas consultas à *API* do Google para verificar a consistência da informação extraída, após a inspeção de um engenheiro do conhecimento esta é adicionada ou não à base de conhecimento, auxiliando assim o trabalho de alimentação da base.

Um projeto derivado do projeto Cyc é o projeto *ThoughtTreasure* (Muller, 2007). Sua base de conhecimento é composta por 35 mil sentenças e palavras da língua inglesa, 21 mil sentenças e palavras da língua francesa, e 27 mil conceitos e 51 mil sentenças de senso comum sobre esses conceitos.

*ThoughtTreasure* adota a abordagem de Minsky (1986), que diz que não há um único tipo de representação que seja o ideal para todos os casos. Por conta disso, o projeto utiliza *grids*, para representar configurações; autômatos finitos, para representar regras, comportamentos de dispositivos e processos mentais; e expressões lógicas para fatos enciclopédicos e conhecimento de lingüística. Os *grids* utilizados pelo projeto são similares a *ASCII art*, onde cada caractere representa um tipo de objeto diferente, como, por exemplo, parede, janela, porta e mesa.

Outra abordagem de construção de bases de conhecimento de senso comum é adotada pelo projeto *Open Mind Common Sense (OMCS)*, criado pelo *Media Lab* no *Massachusetts Institute of Technology (MIT)* (Singh, 2002b). Essa abordagem parte do pressuposto de que qualquer pessoa tem o conhecimento de senso comum necessário para prover senso comum aos computadores. E com isso utiliza uma abordagem colaborativa utilizando um *site* na Internet como meio de capturar conhecimento de senso comum.

O site do projeto está disponível desde setembro de 2000. Nele, qualquer pessoa com acesso à Internet pode entrar e, após um breve cadastro, responder a perguntas sobre os mais diversos assuntos, chamados de atividades, que são disponibilizados através de gabaritos. A cada resposta inserida, o tipo da atividade é alterado aleatoriamente. Se o colaborador desejar, ele poderá escolher uma atividade específica.

Um exemplo de como essas atividades são apresentadas no *site* pode ser observado no gabarito mostrado na Figura 1; que está associado à atividade “Usos e funções” e possui dois campos. Um deles é preenchido automaticamente pelo sistema com um objeto a partir da retroalimentação dos dados, nesse exemplo preenchido com “*A street*”, enquanto o segundo campo deve ser preenchido pelo usuário.

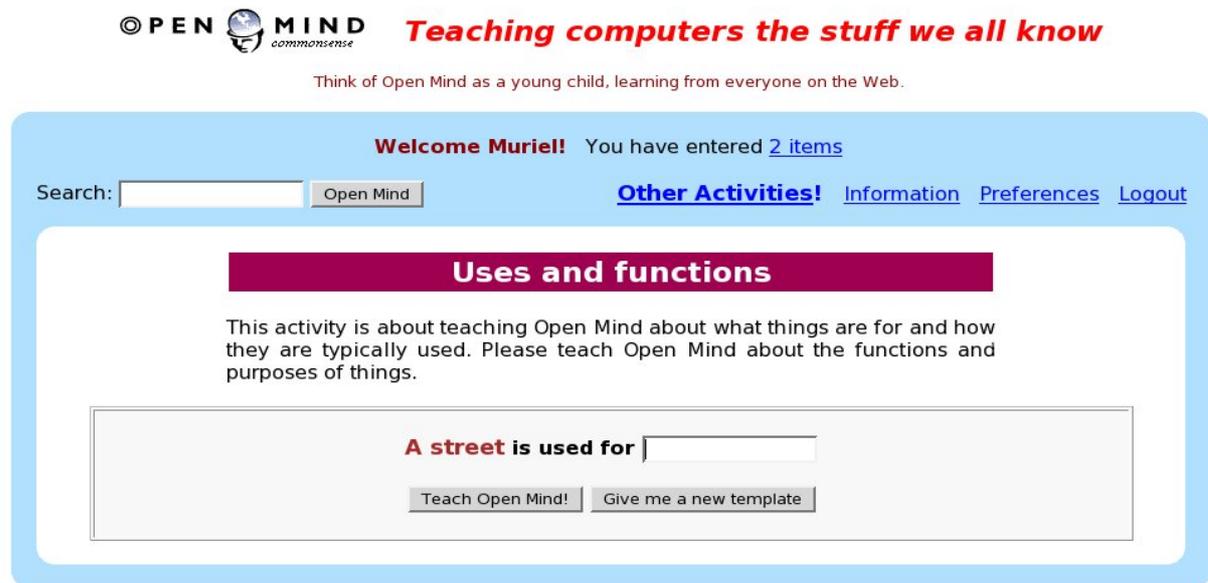


FIGURA 1. Apresentação de um gabarito do site do Projeto OMCS.

A primeira aplicação a fazer uso dessas sentenças empregou regras para a extração do conhecimento e construção de uma rede semântica: o sistema de inferência robusto de senso comum chamado de CRIS (Commonsense Robust Inference System), utilizado pelo software ARIA (Lieberman; Liu, 2002), um sistema de recuperação de fotos. Uma rede semântica (Richens, 1958) é uma forma utilizada para representação do conhecimento utilizando um grafo direcionado, onde os vértices representam conceitos e as arestas representam relações semânticas entre os conceitos. Com ele, extraem-se conhecimentos taxonômicos, espaciais, funcionais, causais e emocionais da OMCS, para criar uma rede semântica e usar a propagação de ativação para melhorar a recuperação de informação.

O método utilizado pelo CRIS para a recuperação de informações sugeriu uma nova abordagem para construir uma base de conhecimento de senso comum. Em vez de construir diretamente a estrutura de conhecimento usada pelo sistema de raciocínio, como é feito no projeto Cyc, no projeto OMCS a entrada de conhecimento é realizada utilizando sentenças em linguagem natural. E dessas sentenças semi-estruturadas extraídas através dos gabaritos, é gerado o conhecimento em uma representação estruturada.

A partir do CRIS, foi construída uma rede semântica chamada OMCSNet, pela

reformulação sistemática de todas as sentenças da OMCS, além da construção de um conjunto de funções para a utilização da OMCSNet, com três funções principais, a saber: encontrar caminho entre nós, capturar contexto e capturar conceitos análogos.

A ConceptNet é a evolução do CRIS/OMCSNet, a versão 2.1 conta com 1,6 milhões de *links* relacionando 300 mil nós (Liu; Singh, 2004). Os nós da rede semântica da ConceptNet são relacionados através relações semânticas. As vinte possíveis relações ilustradas na Tabela 1.

**TABELA 1.** Tipos de relações semânticas da ConceptNet. Adaptada de (Singh, 2002b).

Tipo	Predicado	Exemplo			
		Argumento 1	Argumento 2	f	i
K-lines	ConceptuallyRelatedTo	bad breath	mint	4	0
	ThematicKLine	wedding dress	veil	9	0
	SuperThematicKLine	western civilization	civilization	0	12
Coisas	IsA	horse	mammal	17	3
	PropertyOf	fire	dangerous	16	1
	PartOf	butterfly	wing	5	1
	MadeOf	bacon	pig	3	0
	DefinedAs	meat	flesh of animal	2	1
Agentes	CapableOf	dentist	pull tooth	4	0
Eventos	PrerequisiteEventOf	read letter	open envelope	2	0
	FirstSubeventOf	start fire	light match	2	3
	SubeventOf	play sport	score goal	2	0
	LastSubeventOf	attend concert	applaud	2	1
Espaciais	LocationOf	army	war	3	0
Causais	EffectOf	view video	entertainment	2	0
	DesirousEffectOf	sweat	take shower	3	1
Funcionais	UsedFor	fireplace	burn wood	1	2
	CapableOfReceivingAction	drink	serve	0	14
Afetivas	MotivationOf	play game	compete	3	0
	DesireOf	person	not be depressed	2	0

É interessante observar, que essa versão da ConceptNet possui um sistema para atribuir

pesos aos conhecimentos, que expressa quantas vezes o conhecimento foi encontrado no banco de dados e quantas vezes o conhecimento pôde ser inferido a partir de outros conhecimentos. Esses pesos também podem ser observados na Tabela 1 nas colunas f e i respectivamente.

As relações do tipo *K-line (Knowledge Line)* foram criadas com base na teoria da memória de Minsky (1979). Segundo essa teoria, a função da memória é recriar um estado da mente. Quando se aprende ou memoriza algo, uma k-line é criada; essa k-line é conectada aos agentes mentais que estão ativos no momento do evento do aprendizado, e quando essa k-line é ativada, ela ativa esses agentes mentais, recriando um “estado mental parcial” parecido com o estado original.

Essas relações destacam a importância do contexto, unificando temas e suas variações como, por exemplo: “*buy food*” e “*purchase groceries*”. Isso possibilita uma maior conectividade entre os nós e a geração de variações do nó, considerando as variações lingüísticas presentes nos textos do mundo real. Atualmente, 1,25 milhões das 1,6 milhões de relações da rede semântica são do tipo k-line, o que reforça a importância das relações desse tipo.

A construção da *ConceptNet* é realizada em três fases. Na primeira, a fase de extração, aproximadamente cinquenta expressões regulares e regras sintáticas e semânticas são usadas para extrair relações binárias das sentenças da base de dados OMCS. Essa extração é facilitada pela maneira com que as sentenças são inseridas no *site*, pelo preenchimento dos gabaritos, capturando sentenças semi-estruturadas em inglês. Por exemplo, “*The effect of [falling off a bike] is [you get hurt]*”. Sentenças para as quais não haja nenhum tipo de relação ainda podem ser extraídas pela relação *K-line* genérica “*ConceptuallyRelatedTo*” se elas contêm termos semânticos que possam ser utilizados.

Na segunda, a fase de normalização, os nós extraídos são normalizados pelo

*MontyLingua* (Liu, 2004). Erros ortográficos são corrigidos por um corretor ortográfico automático. São removidos das construções sintáticas os artigos, verbos modais e quaisquer outros acompanhantes. Os verbos são transformados na sua forma infinitiva.

Na terceira, a fase de relaxamento, possui um conjunto de heurísticas de relaxamento produzindo um conhecimento adicional intermediário, aumentando a conectividade da rede. Uma heurística é uma técnica de solução de problemas que ignora se a solução pode ser provada, mas que produz, na maioria das vezes, bons resultados. (Pearl, 1984). A seguir é apresentada uma breve visão sobre as heurísticas utilizadas:

- Nós duplicados são fundidos, gerando um metacampo frequência em cada predicado, para armazenar quantas vezes esse conhecimento foi expresso.
- Relações do tipo “*IsA*” são usadas para levar heurísticamente conhecimento dos nós filhos para os nós pais.
- Generalizações léxicas e temáticas são produzidas relacionando conhecimentos mais específicos aos mais genéricos, produzindo relações do tipo “*SuperThematicKLine*”.

São exemplos desse tipo de relação:

```
(SuperThematicKLine "buy food" "buy")
```

```
(SuperThematicKLine "purchase food" "buy")
```

- Frases substantivas que contêm adjetivos são refinadas em uma relação adicional do tipo “*PropertyOf*”. Por exemplo: “(*IsA 'apple' 'red round object'*)” e “(*IsA 'apple' 'red fruit'*)” implicam em “(*PropertyOf 'apple' 'red'*)”.
- Diferenças de vocabulário como “*bike*” e “*bicycle*”, variações morfológicas como “*relax*” e “*relaxation*” são ligadas pela adição de uma relação do tipo “*SuperThematicKLine*”.

Os conhecimentos gerados durante esse passo são responsáveis pelo peso *i* presente na estrutura da *ConceptNet*, registrando quantas vezes o mesmo conhecimento foi inferido, de

forma a mensurar o peso da relação.

## **2.4. O Projeto Open Mind Common Sense no Brasil**

O Projeto Open Mind Common Sense no Brasil - OMCS-Br foi criado dentro do contexto do Projeto *Multi-Lingual Common Sense* (Lieberman; Schmandt, 2005), que estimula a criação de bases de senso comum em outros idiomas além do inglês, visando ao aumento do entendimento e cooperação entre os povos de diferentes culturas e idiomas.

O Laboratório de Interação Avançada (LIA) da Universidade Federal de São Carlos (UFSCar) é responsável pela variante brasileira do projeto. Em consequência disso, os pesquisadores do LIA, começaram seu trabalho de construir uma base de senso comum em língua portuguesa, adaptando os elementos providos pela arquitetura do projeto OMCS original e desenvolvendo novos quando necessário.

Desde então, no LIA tem-se procurado, além dos objetivos originais do projeto *Multi-Lingual Common Sense*, explorar as possibilidades de utilização de bases de conhecimento de senso comum em outras áreas de pesquisa, tais como: Interação Humano-Humano (IHH); e na informática na educação, focando principalmente a Aprendizagem Eletrônica (AE).

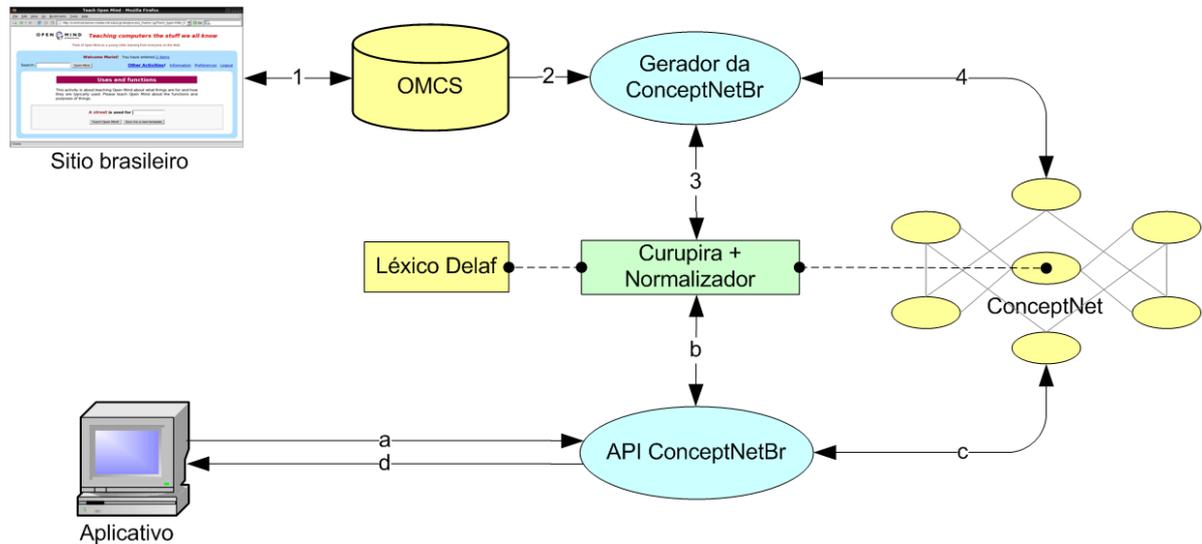
Em relação à IHH, está sendo explorada a possibilidade do uso do conhecimento de senso comum em softwares que apoiem a comunicação entre pessoas de diferentes culturas, alertando e evitando possíveis desentendimentos (Anacleto, 2006).

Em relação à informática na educação, além do trabalho que explora a geração de analogias para materiais instrucionais, tem-se explorado o uso do conhecimento de senso comum como apoio ao professor durante o planejamento de uma Ação de Aprendizagem (AA) e o uso do senso comum no desenvolvimento de jogos educacionais (Ferreira, 2007).

### **2.4.1. A arquitetura do projeto OMCS-Br**

A arquitetura desse projeto é similar à arquitetura utilizada no projeto OMCS. Foram, contudo, realizadas modificações para contemplar as diferenças entre a língua inglesa e

portuguesa. No diagrama da arquitetura, que está ilustrado na **Figura 2**, podem ser observadas duas partes:



**Figura 2.** Diagrama da arquitetura do projeto OMCS-Br.

A primeira parte é composta por um sitio no qual (1) dados são inseridos em linguagem natural por meio de gabaritos, Esses dados são armazenados na base, para retroalimentar novos gabaritos. Esse ciclo pode se repetir várias vezes durante a captura pelo sitio; (2) em seguida os dados capturados são enviados ao gerador da *ConceptNetBr*; (3) esses dados são submetidos ao analisador sintático Curupira, que adiciona as etiquetas morfológicas e são normalizados pelo módulo Normalizador, que os envia de volta ao gerador da *ConceptNetBr*; (4) com os dados normalizados, o gerador cria um conjunto de relações iniciais e, consultando essas relações iniciais, o gerador aplica um conjunto de heurísticas para aumentar a conectividade da rede, criando novas conexões entre os nós existentes (4).

Na segunda parte, (a) o aplicativo que faz uso de senso comum envia seus dados em modo texto para a *API (Application Program Interface)* da *ConceptNetBr*; (b) a *API* envia esses dados para o analisador sintático Curupira e ao módulo Normalizador, que os devolve à *API*; (c) os dados normalizados são utilizados na busca de conhecimentos de senso comum na *ConceptNetBr*; (d) o conhecimento obtido na busca é enviado de volta à aplicação, que usa esse conhecimento conforme seu interesse e necessidade.

Na próxima seção será explicado com detalhes cada um desses módulos presentes na arquitetura da versão brasileira do projeto.

### 2.4.2. A Coleta dos Dados

Um sítio, equivalente ao do Projeto OMCS original, foi desenvolvido e está disponível desde setembro de 2005 e até julho de 2007 obteve cerca de 160 mil entradas, neste projeto chamadas de fatos. O sítio está hospedado no servidor do LIA, rodando sob o sistema operacional *Linux*, distribuição *Slackware*. O site foi desenvolvido utilizando a tecnologia *Java Server Pages* (JSP) sob o servidor WEB *Apache Tomcat*, e armazena as informações utilizando o banco de dados relacional *MySql*.

Além das 20 atividades disponibilizadas que foram baseadas no sítio do projeto original, foram implementadas três atividades temáticas focadas nos interesses dos pesquisadores do LIA, com o objetivo de acelerar a coleta de dados sobre domínios específicos. Pode-se observar abaixo na Figura 3, um exemplo de um gabarito para coletar dados referentes ao que se pode fazer com alguns materiais. As trechos “Você pode utilizar” e “para fazer um(a)” são fixos, e o trecho “ouro” é dinâmico, assim cabe ao colaborador preencher o último campo.

© PEN MIND *Ensinando ao computador as coisas que todos nós sabemos*  
common sense no Brasil Você é um pesquisador? Faça o download do Open Mind Common Sense hoje!

Bem vindo Muriel de Souza Godoil! 799 aceitos / 15 em revisão

Busca:   [Outras atividades!](#) [Informação](#) [Revisão!](#) [Atualize seus dados](#) [Sair](#)

**Feito de**

Fale sobre o material do qual as coisas são feitas **Novo!**

Você pode utilizar **ouro** para fazer um(a)

FIGURA 3. Exemplo de gabarito utilizado para coletar dados no site do projeto OMCS-Br.

A parte dinâmica é gerada automaticamente, a partir das entradas de outros colaboradores. Assim, uma resposta fornecida por um colaborador pode ser utilizada para

gerar uma nova pergunta. Por exemplo, na Figura 3 a palavra ouro foi a resposta oferecida por algum colaborador anterior e foi usada para compor o novo gabarito. Outro exemplo de retroalimentação pode ser observado na Figura 4, onde a entrada de um usuário no gabarito referente à localização é utilizada em outro gabarito para coletar dados referentes a usos.

The image shows two overlapping form panels. The top panel, titled 'Localização', has a purple header and the instruction 'Descreva onde as coisas são tipicamente encontradas'. Below this, it says 'Você geralmente encontra um(a)  em um(a) cadeira'. There are buttons for 'Ensinar!' and 'Isso não faz sentido'. The bottom panel, titled 'Usos', has a purple header and the instruction 'Descreva como objetos são usados'. It shows 'Um(a) parafuso é usado(a) para '. At the bottom of this panel are buttons for 'Ensinar!', 'Isso não faz sentido', 'Pular', and 'Atividades aleatórias'. A red arrow points from the 'parafuso' input in the 'Localização' form to the 'parafuso' input in the 'Usos' form.

FIGURA 4. Exemplo de retroalimentação para geração de novas templates.

O sistema de retroalimentação é importante por permitir uma maior variação dos temas apresentados. Em um questionário aplicado aos colaboradores do site, muitos deles mencionaram que os gabaritos estáticos são mais chatos de serem preenchidos, o que pode desestimular o trabalho dos colaboradores. Esse fato é importante para o projeto OMCS-Br que depende de colaboradores voluntários para construir a base de conhecimento de senso comum necessária para a realização das pesquisas.

O conjunto de gabaritos, que compõem as atividades gerais disponibilizadas no site, pode ser observado na Tabela 2. Os campos dos gabaritos sinalizados por ( ) representam campos que são preenchidos automaticamente através da retroalimentação. Os campos dos gabaritos sinalizados por [ ] representam campos a serem preenchidos pelo usuário do *site*.

TABELA 2. Conjunto de atividades e gabaritos disponibilizados no sítio OMCS-Br.

Atividades	Templates
Ajuda	Quando pessoas ( ) uma forma de ajudar é [ ]
Classificação	Um (a) ( ) é um tipo de [ ]
Coisas	Uma coisa que você pode encontrar em um(a) ( ) é um(a) [ ]. Uma coisa que você (frequência <sup>1</sup> ) encontra em um(a) ( ) é um(a) [ ]
Definições	Um(a) ( ) pode ser definido como [ ] ( ) pode ser uma definição de um(a) [ ]
Efeito de	( ) pode ser um efeito de [ ] Um efeito de ( ) pode ser [ ]
Eventos	A primeira coisa que deve ser feita para ( ) é [ ] A última coisa que deve ser feita para ( ) é [ ] Alguns passos que devem ser feito para ( ) são: 1. [ ], 2. [ ], 3. [ ] 4. [ ], 5. [ ]
Feito de	Você pode utilizar um(a) ( ) para fazer um(a) [ ] Um material utilizado para fazer um(a) ( ) é [ ]
Habilidades	( ) é uma habilidade de um(a) [ ] Um(a) ( ) é capaz de [ ]
Imagem	(Imagem): 1. [ ], 2. [ ], 3. [ ], 4. [ ], 5. [ ].
Localização	Você ( ) encontra um(a) ( ) em um(a) [ ] Você ( ) encontra um(a) [ ] em um(a) ( )
Paráfrase	Um outro jeito de dizer ( ) é [ ]
Partes	Um(a) ( ) pode ser uma parte de um(a) [ ] Um(a) ( ) pode ser formado das seguintes partes: 1. [ ], 2. [ ], 3. [ ], 4. [ ], 5. [ ]
Personalidades	A pessoa/personagem ( ) é/foi [ ]. Algumas coisas que ele(a) faz/fez são: 1. [ ], 2. [ ], 3. [ ] A pessoa/personagem ( ) é/foi ( ). Algumas coisas que ele(a) faz/fez são: 1. [ ], 2. [ ], 3. [ ] A pessoa/personagem ( ) é/foi ( ). Uma coisa que ele(a) faz/fez é [ ]
Pessoas	Pessoas [ ] quando elas [ ]
Problemas	Quando se tenta ( ) um problema encontrado pode ser [ ]
Propriedades	Cite alguns adjetivos para ( ): 1. [ ], 2. [ ], 3. [ ], 4. [ ], 5. [ ]. ( ) pode ser uma das características de um(a): 1. [ ], 2. [ ], 3. [ ], 4. [ ], 5. [ ]
Sentenças	Você (frequência) quer um(a) ( ) para [ ]
Situações	Um(a) [ ] pode querer [ ], quando [ ] Um(a) ( ) pode querer [ ], quando [ ] Um(a) [ ] pode querer ( ), quando [ ] Um(a) [ ] pode querer [ ], quando ( )
Usos	Um(a) ( ) é usado para [ ]

Os seguintes gabaritos apresentam uma lista de três a cinco campos numerados para serem preenchidos, a saber, “eventos”, “imagens”, “partes”, “personalidades” e

<sup>1</sup> Termo que denota frequência, e corresponde a um dos seguintes valores: quase nunca, raramente, geralmente não, às vezes, geralmente, frequentemente, muito frequentemente, quase sempre

“propriedades”. Isto não significa que o usuário tenha que necessariamente preenchê-los todos. Porém cada campo preenchido é contado como uma contribuição diferente. Essa abordagem adotada pelo projeto brasileiro também reflete opiniões dadas por usuários como um meio de facilitar a coleta. Um exemplo desse tipo de gabarito pode ser observado na Figura 5.

© PEN MIND *common sense no Brasil* **Ensinando ao computador as coisas que todos nós sabemos**  
 O Open Mind será usado para fazer computadores mais inteligentes e amigáveis.

Bem vindo Muriel de Souza Godoi! 799 aceitos / 15 em revisão

Busca:   [Outras atividades!](#) [Informação](#) [Revisão!](#) [Atualize seus dados](#) [Sair](#)

**Partes**

Conte-nos sobre as partes que compõem os objetos!

Você pode indicar mais de uma parte por vez, preenchendo os espaços em branco numerados de 1 a 5, com uma parte diferente do objeto indicado.

Use a tecla "Tab" para passar de um campo para o outro e quando quiser enviar os dados pressione a tecla "Enter" ou clique no botão "Ensinar". Cada campo preenchido será contado como uma contribuição.

Um(a) **embalagem** pode ser uma parte de um(a):

1.
2.
3.
4.
5.

FIGURA 5. Gabarito exibindo cinco campos para facilitar o trabalho dos colaboradores.

Uma modificação realizada pelos pesquisadores do projeto OMCS-Br no processo de retroalimentação dos dados foi o desenvolvimento de um mecanismo para a revisão humana dos fatos colhidos. Essa revisão é realizada pelos revisores do laboratório LIA, antes de serem utilizados para a retroalimentação, devido à ausência de um corretor ortográfico não supervisionado. Essa visão minimiza os efeitos de uma pequena minoria de dados inseridos incorretamente atrapalharem a coleta de novos dados, como por exemplo, usuários mal intencionados, erros ortográficos e entradas sem sentido.

Vale ressaltar, no entanto, que os revisores em nenhum momento avaliam a semântica

das sentenças que estão sendo revisadas, esse fator diferencia a abordagem deste projeto da abordagem de outros projetos, como Cyc. Considera-se que, se o revisor julgar a semântica das sentenças, ele estará, de certa forma, utilizando o seu próprio senso comum para fazer esse julgamento e influenciando, assim, o senso comum armazenado na base. Essa influência não é desejada neste projeto, onde se considera que o conhecimento de senso comum não reflete, necessariamente, a verdade científica, mas reflete o conhecimento compartilhado por um grupo de pessoas em um determinado lugar e em uma determinada época. Uma tela do sistema de revisão desenvolvido pode ser observada na Figura 6.

© PEN MIND *Ensinando ao computador as coisas que todos nós sabemos*  
common sense no Brasil Por favor, conte aos seus amigos sobre nós!

Bem vindo Muriel de Souza Godoi! 799 aceitos / 15 em revisão

Busca:   [Outras atividades!](#) [Informação](#) [Revisão!](#) [Atualize seus dados](#) [Sair](#)

### Revisar contribuições

Você revisou 16628 entradas.

Aceitar	Manter	Rejeitar	Conhecimento	Retro-Alimentar	Data
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) abobrinha é usado(a) para comer	<input checked="" type="checkbox"/>	2007-06-12 11:56:29.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) açúcar é usado(a) para adoçar	<input checked="" type="checkbox"/>	2007-02-28 14:38:33.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) aeronave é usado(a) para voar	<input checked="" type="checkbox"/>	2007-02-27 16:52:15.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) air bag é usado(a) para evitar lesões graves	<input checked="" type="checkbox"/>	2007-05-27 03:29:59.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) alimentação balanceada é usado(a) para manter-se saudavel	<input checked="" type="checkbox"/>	2007-02-28 14:39:07.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) ampola é usado(a) para Armazenar	<input checked="" type="checkbox"/>	2007-02-27 16:48:47.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) anão de cimento é usado(a) para enfeitar o jardim	<input checked="" type="checkbox"/>	2007-02-28 15:12:30.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) anotação é usado(a) para ser esquecida	<input checked="" type="checkbox"/>	2007-03-20 23:36:58.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) apartamento é usado(a) para residir	<input checked="" type="checkbox"/>	2007-02-28 14:29:36.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) baiano é usado(a) para trabalhar na construção civil	<input checked="" type="checkbox"/>	2007-04-13 10:10:36.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) banco de supino é usado(a) para apoiar a barra	<input checked="" type="checkbox"/>	2007-02-28 15:04:04.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) bandeja com sonhos é usado(a) para servir aos convidados	<input checked="" type="checkbox"/>	2007-04-13 10:10:20.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) barbeador é usado(a) para aparar pelos	<input checked="" type="checkbox"/>	2007-02-28 14:50:52.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) barraca de frutas é usado(a) para comercializar frutas	<input checked="" type="checkbox"/>	2007-02-28 14:06:03.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) barraca de peixes é usado(a) para vender	<input checked="" type="checkbox"/>	2007-06-12 11:56:36.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) batata é usado(a) para fazer babata frita, cozida, sotê	<input checked="" type="checkbox"/>	2007-02-28 15:10:29.0
<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Um(a) brasa é usado(a) para manter quente	<input checked="" type="checkbox"/>	2007-02-27 16:53:54.0

FIGURA 6. Sistema de revisão dos fatos coletados.

O sistema de revisão foi incorporado ao sítio, e está disponível somente aos revisores do projeto. Nele, fatos coletados são exibidos agrupados por atividades, para facilitar o trabalho de revisão. Os revisores têm as seguintes opções: aceitar o fato para base de senso comum e para a retroalimentação; aceitar o fato para a base de senso comum, mas não para a retroalimentação; rejeitar o fato tanto para a base quanto para a retroalimentação.

Pode-se observar um exemplo de erro ortográfico durante o preenchimento dos gabaritos no penúltimo fato exibido na Figura 6, a saber, “babata frita”, neste caso o revisor deve rejeitar o fato. Adicionalmente, foi disponibilizada a opção “Manter” caso o revisor fique em dúvida se deve, ou não, aceitar um fato.

As regras utilizadas no processo de revisão são as seguintes:

- Entradas contendo palavras de baixo calão são rejeitadas para a retroalimentação, porém são aceitas para a base de senso comum, pois se considera que essas palavras podem fazer parte do vocabulário de um determinado grupo de colaboradores.
- Entradas compostas por palavras que não fazem sentido na língua portuguesa como, por exemplo, “jttrhhtrh” são rejeitadas tanto para a base, quanto para a retroalimentação.
- Entradas contendo palavras com erros ortográficos como “cdeira” são descartadas tanto para retroalimentação, quanto para a base.
- Entradas que não correspondem ao padrão esperado pelo gabarito são aceitos para a base, porém não são usadas para a retroalimentação. Em um gabarito que espera um verbo no infinitivo, por exemplo, “*Um(a) cadeira é usado(a) para \_\_\_\_*”, pode ser preenchido com “*conforto*”. Neste caso, essa entrada é aceita na base, mas deve não ser utilizada para a retroalimentação, pois caso seja utilizado para gerar um gabarito que precisa de um verbo ficaria sem sentido; por exemplo, na atividade Ajuda com o gabarito “*Quando se tenta conforto, um problema encontrado pode ser \_\_\_\_\_*”.

O sítio brasileiro também se diferencia do sítio do projeto original por armazenar mais dados referentes ao perfil do colaborador, como gênero, idade, escolaridade, cidade e estado. Assim, pode-se filtrar os dados desejados a fim de levantar o conhecimento de senso comum dos colaboradores com um determinado perfil.

O perfil dos 1185 colaboradores do projeto brasileiro, que é composto por cerca de

80% de homens e 20% de mulheres. Levando em consideração a localização, 73% dos colaboradores são residentes da região sudeste do Brasil, e 15% da região Sul. Isso reflete as áreas mais desenvolvidas do Brasil, onde há mais pessoas com acesso à Internet.

A escolaridade dos colaboradores pode ser observada na Tabela 3, mostra que quase 65% dos colaboradores do site têm ensino superior completo.

**TABELA 3.** Distribuição dos colaboradores por escolaridade.

Escolaridade	Porcentagem
Ensino Básico	2,21 %
Ensino Fundamental	18,17 %
Ensino Superior	65,86 %
Especialização	4,52 %
Mestrado	7,04 %
Doutorado	2,21 %

A maioria dos colaboradores está na faixa de 18 a 29 anos, conforme pode ser observado na Tabela 4.

**TABELA 4.** Distribuição dos colaboradores por faixa etária.

Faixa Etária	Porcentagem
Menos de 13 anos	0,75 %
De 13 a 17 anos	20,51 %
De 18 a 29 anos	67,36 %
De 30 a 45 anos	9,88 %
De 46 a 65 anos	1,22 %
Acima de 65 anos	0,28 %

A fim de aumentar o número de contribuições no sítio, desde seu lançamento os pesquisadores do LIA têm procurado meios de divulgá-lo para o maior número possível de pessoas. Para isso, foram realizadas palestras em algumas Instituições de Ensino Superior do estado de São Paulo, como USP, UNESP, UNICAMP e FABAN, além da divulgação do projeto em jornais e revistas, como os jornais “O Estado de São Paulo”, “Correio Brasiliense” e “A Tribuna”, uma nota na revista “Galileu” e uma reportagem no Jornal EPTV, emissora da região de Campinas, filiada à Rede Globo de Televisão.

Outra atividade desenvolvida para promover o sítio foi a realização de “desafios”,

onde cada desafio é uma espécie de competição em que foram oferecidos prêmios como canetas e camisetas aos três colaboradores que entrassem com o maior número de fatos válidos em determinados prazos. Os desafios foram divulgados através de e-mails aos colaboradores do sítio, além de divulgação no sítio “UEBA” (<http://www.ueba.com.br>), um sítio que disponibiliza diariamente uma lista de sítios considerados *interessantes*.

Na Figura 7, pode-se observar o número de fatos coletados em cada mês do projeto desde seu lançamento. Houve um grande número de colaborações nos dois primeiros meses, do projeto após a divulgação no jornal “O Estado de São Paulo”; os outros três picos de coleta representam os três “desafios” realizados pelo projeto. Isso mostra a grande importância da divulgação do sítio nesse tipo de projeto colaborativo.

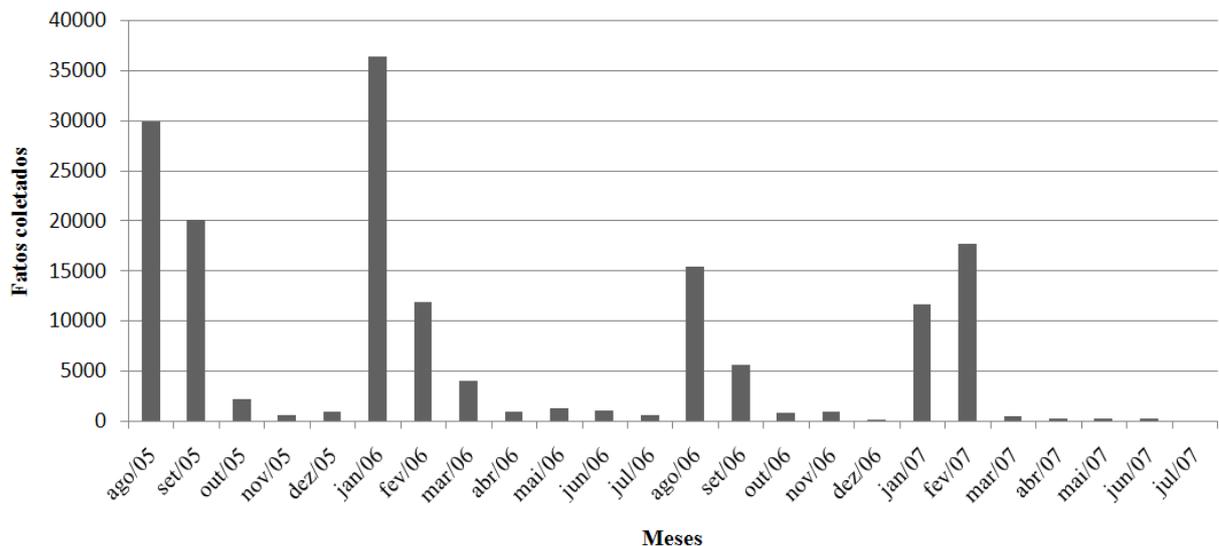


FIGURA 7. Distribuição do número de fatos coletados por mês.

#### 2.4.3. A Construção da *ConceptNetBr*

A *ConceptNetBr* é a versão brasileira da rede semântica *ConceptNet*, a qual foi utilizada como ponto de partida para o desenvolvimento da versão brasileira. Ela utiliza as mesmas 20 relações apresentadas na Tabela 1, do projeto OMCS.

A *ConceptNetBr* é armazenada em arquivos quatro arquivos no formato texto. Nesses arquivos, cada linha representa uma relação entre dois conceitos no seguinte formato:

(Predicado "argumento1" "argumento2" "f=m;i=n")

O predicado representa a relação entre os dois argumentos; a variável  $f$  que representa o número de vezes que o conhecimento foi obtido a partir das regras de extração, ou seja, através do sítio; a variável  $i$  representa o número de vezes que o conhecimento foi obtido a partir das regras de inferência. Um trecho do arquivo utilizado pela *ConceptNetBr* pode ser observado na Figura 8, a seguir.

```
(LocationOf "caixa de morango" "quitanda" "f=3;i=0")
(UsedFor "restaurante" "almoçar" "f=2;i=0")
(CapableOf "criança" "aprender" "f=7;i=0")
(MadeOf "piscina" "azulejo" "f=2;i=0")
(IsA "ornitorrinco" "animal" "f=2;i=1")
(MotivationOf "cuidar" "estar doente" "f=3;i=0")
(PropertyOf "chá" "quente" "f=1;i=4")
(PartOf "gato" "focinho" "f=2;i=0")
```

FIGURA 8. Exemplo do arquivo de dados utilizado pela *ConceptNetBr*.

Três passos são necessários desde a exportação dos dados do sítio até a obtenção dos dados em um formato que torne possível a utilização em aplicações computacionais: extração, normalização e relaxamento.

Antes de iniciar o processamento dos fatos coletados pelo sítio, um *script* desenvolvido em PHP é responsável por exportar os dados do banco de dados *MySQL* para um arquivo texto, juntamente com o perfil do usuário que colaborou, possibilitando assim a utilização do senso comum dos colaboradores de um determinado perfil, utilizando os caracteres “\$\$” como separadores de campo. O arquivo resultante deste *script* é então submetido ao primeiro passo: a extração.

A seguir, a Seção 2.4.3.1 descreve o primeiro passo da construção da *ConceptNetBr*, a extração. O segundo passo, a normalização, é apresentado na Seção 2.4.3.2. A construção é finalizada no terceiro passo, abordado na Seção 2.4.3.3.

#### 2.4.3.1. Extração

Um conjunto de 81 regras possibilita transformar os fatos em linguagem natural semi-estruturada obtidos do sítio para a forma de predicados. Essas regras foram desenvolvidas

utilizando expressões regulares. Quando uma regra identifica o padrão é produzida a saída correspondente. Um exemplo de regra de extração e sua utilização pode ser observada na Figura 9: (1) tem-se o fato do sítio a ser processado, seguido pelos dados do perfil do colaborador; (2) a entrada é submetida a um conjunto de expressões regulares, até ocorrer o casamento com uma regra que é responsável por um determinado gabarito; (3) o módulo de extração então gera a saída correspondente, salvando a saída em outro arquivo, que será utilizado como entrada do próximo passo, a saber, a normalização.

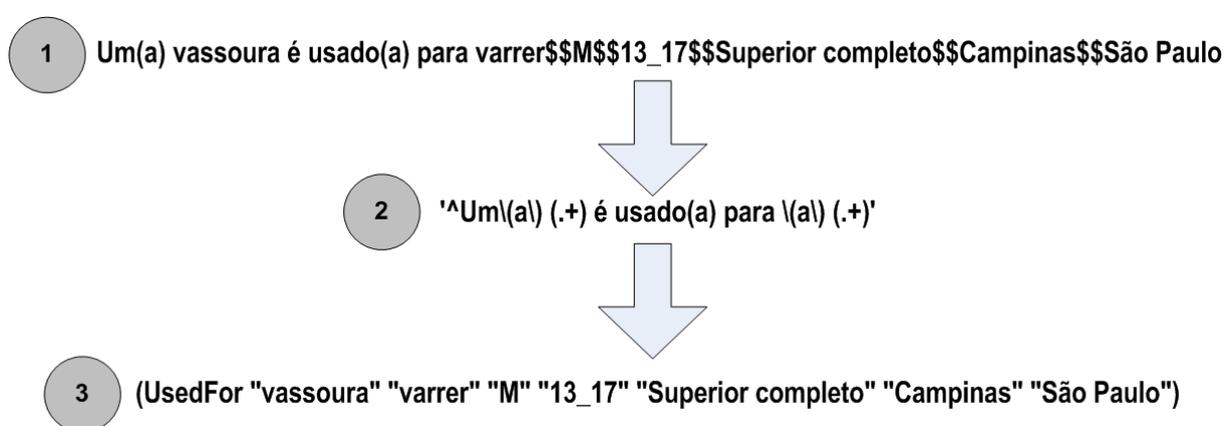


FIGURA 9. Fase de extração durante a geração da *ConceptNetBr*.

#### 2.4.3.2. Normalização

Durante a coleta dos dados, os colaboradores podem digitar a mesma resposta de diferentes maneiras, considerando artigos, flexões de verbos, complementos, entras formas. Faz-se necessário, portanto, um processo que unifique essas diferentes variações para a geração de uma rede semântica mais coesa. Esse processo de unificação é chamado, no contexto do projeto OMCS-Br, de normalização.

No projeto OMCS, a normalização é realizada pelo MontyLingua, um analisador sintático de linguagem natural para a língua inglesa, que classifica os tokens das sentenças com etiquetas morfológicas e retira as inflexões morfológicas das sentenças capturadas pelo site. No entanto, esse analisador sintático não pode ser utilizado para a normalização de sentenças em Português, devido às diferenças léxicas e morfológicas existentes entre essas





Um esquema geral que mostra o processo de normalização aplicado a uma sentença pode ser observado na Figura 11. A entrada mostrada em (1) é etiquetada pelo analisador sintático Curupira resultando em (2), e depois, o Módulo Normalizador é responsável por substituir cada *token* pela sua forma canônica correspondente conforme mostrado em (3).

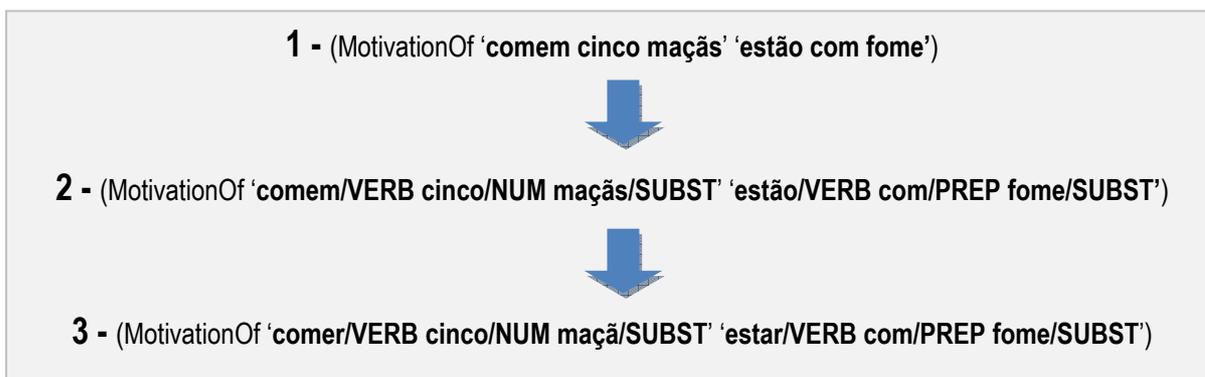


FIGURA 11. Processo de normalização dos argumentos das relações.

Vale ressaltar que, nessa figura, suprimiram-se os dados referentes ao perfil do colaborador em troca de uma melhor explicação, visto que o foco desta seção está nos textos das entradas e não no perfil do colaborador; porém esses dados estão presentes no arquivo de entrada e saída do processo de normalização.

Testes foram realizados a fim de mensurar os efeitos do processo de normalização na rede semântica gerada. Foram geradas duas redes, uma utilizando a normalização e outra sem a normalização, e calculados o número de nós e vértices distintos e com isso se chegou à densidade média de vértices por nó. Isso tudo pode ser observado na Tabela 5.

TABELA 5. Efeitos da normalização na rede semântica gerada.

	Sem a normalização	Com a normalização	Normalizado / Não normalizado
Nós distintos	36.219	31.423	-13,94 %
Vértices distintos	61.445	57.801	-5,95 %
Densidade média de vértices por nó	3,3939	4,4643	+31,57 %

Observa-se que, com a normalização, diminuiu-se em 13,94% o número de nós e em 5,95% o número de vértices únicos presentes na rede, confirmando a tendência da normalização em unificar as variações morfológicas. Com isso ocorreu um aumento da

conectividade da rede gerada, o que pode ser observado pelo aumento de 31,57% na densidade média de vértices por nó. Ou seja, houve um aumento na média do número de vértices que estão ligados a cada nó, gerando uma rede mais coesa e conexa.

#### **2.4.3.3. Relaxamento**

Na fase de relaxamento, novas relações são geradas através da aplicação de regras de inferências, baseadas em heurísticas, às relações obtidas através da normalização. Esse processo também visa aumentar a conectividade da rede gerada. A maior parte das relações geradas são do tipo *K-line*, que visam a ajudar a *ConceptNetBr* mapear melhor as variações presentes em textos do mundo real. Essas relações são baseadas na teoria da memória de Minsky (1979). As heurísticas utilizadas, juntamente com seus respectivos exemplos, são listadas a seguir:

- Relações com predicado do tipo *SuperThematicKLine*, que representam relações de generalização e especialização entre conceitos, são geradas a partir da “quebra” de entradas composta por várias palavras. Por exemplo, com a quebra da entrada “cachorro grande” são geradas as relações (*SuperThematicKLine* “cachorro grande” “cachorro”) e (*SuperThematicKLine* “cachorro grande” “grande”).
- Relações do tipo *IsA* com um adjetivo no segundo termo geram relações do tipo *PropertyOf*. Por exemplo, (*IsA* “casa” “pequena”) gera a relação (*PropertyOf* “casa” “pequena”).
- Quando nos textos das entradas é encontrado um substantivo seguido de verbo é gerada uma relação *CapableOf*. Por exemplo, “garçom servir” gera a relação (*CapableOf* “garçom” “servir”).
- Uma relação do tipo *CapableOfReceivingAcion* é gerada quando é encontrado um verbo seguido de substantivo. Por exemplo, quando é encontrado “pentear cabelo” em uma das entradas gera a relação (*CapableOfReceivingAcion* “cabelo” “pentear”).

Uma última heurística é aplicada para generalizar relações do tipo *PropertyOf* de conceitos filhos para conceitos pais. Conforme pode ser observado na Figura 12, onde a relação (*PropertyOf* "fruta" "doce") é inferida após a análise que várias frutas têm a propriedade de serem doces.

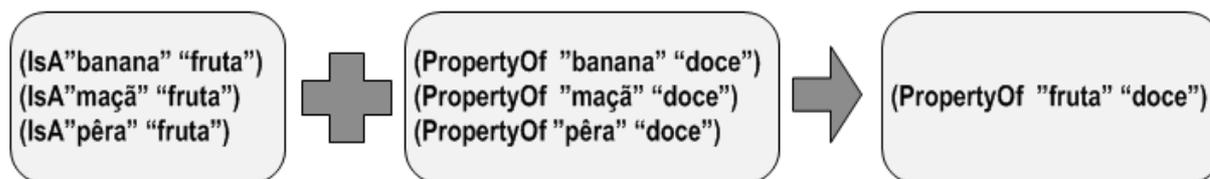


FIGURA 12. Levando conhecimento do tipo *PropertyOf* dos nós filhos para nós pais.

Porém, como ela depende de mais de uma relação para ser gerada, não se pode aplicar essa heurística antes que seja definido o perfil de colaboradores desejado. Assim, essa última heurística será aplicada no momento do carregamento da *ConceptNetBr*, e foi explicada junto com as outras somente com a finalidade de facilitar o entendimento geral das heurísticas.

#### 2.4.4. Utilização da *ConceptNetBr*

Uma vez gerada, a *ConceptNetBr* é armazenada em um arquivo texto, contendo as relações extraídas do sítio já normalizadas e as relações adicionais geradas durante a fase de relaxamento. Faz-se necessário, portanto, um processo de carga desses dados antes que eles sejam utilizados pelas aplicações.

Durante a instanciação, caso desejado, a *ConceptNetBr* pode receber os parâmetros do perfil que devem ser considerados para o carregamento dos dados, incluindo gênero, faixa etária, escolaridade, cidade e estado, que então os envia para um módulo, chamado de Filtragem, que é responsável por selecionar os dados que correspondem ao perfil solicitado e calcular os pesos *f* e *i* de cada relação.

O módulo de Filtragem utiliza um sistema de *cache* para agilizar o carregamento da *ConceptNetBr*. Assim, se um mesmo perfil estiver sendo instanciado pela segunda vez, o filtro somente carregará o arquivo correspondente processado anteriormente; caso contrário,

processa toda a base de dados aplicando o filtro, e aplicará a última heurística explicada na seção anterior, calculará os pesos  $f$  e  $i$  de cada relação e salvará o resultado nos arquivos correspondentes àquele perfil. Os nomes dos arquivos de saída correspondem ao filtro aplicado. Assim, é possível ter diferentes redes semânticas, cada uma representando o conhecimento de senso comum dos colaboradores de um determinado perfil.

Com os arquivos de dados prontos, a *ConceptNetBr* carregará as relações, transformando os argumentos das relações em inteiros para agilizar as buscas e minimizar o uso de memória. As relações são representadas em duas tabelas *hash*, indexadas pelo primeiro e segundo argumentos, com o intuito de aumentar a eficácia das buscas.

A seguir, a Seção 2.4.4.1 apresenta as funções presentes na *ConceptNetBr*. A seção 2.4.4.2 apresenta o módulo servidor, que permite a execução remota das funções da API.

#### **2.4.4.1. Funções da API *ConceptNetBr***

A API da *ConceptNetBr* é a interface de comunicação entre as aplicações que fazem uso do conhecimento de senso comum e a rede semântica. Ela é composta por oito métodos desenvolvidos dentro pelo projeto *OMCS*, com pequenas adaptações para o projeto brasileiro. A principal delas é referente à utilização do Curupira em vez do MontyLingua.

Quatro funções recebem como entrada conceitos normalizados, que são utilizados diretamente para realizar consultas específicas na rede semântica. Essas quatro funções permitem o seguinte:

- Obter todas as relações que têm um determinado conceito como argumento;
- Capturar a lista de conceitos que pertencem ao contexto de um determinado conceito buscado, essa captura é realizada utilizando a técnica clássica da IA de ativação por espalhamento (*spread activation*);
- Capturar projeções de: causas/conseqüências, propriedades, espaciais e efetivas referentes ao conceito buscado, isso é realizado através de variações do algoritmo de

captura de contexto, levando em consideração somente determinados tipos de relações;

- Descobrir conceitos análogos ao conceito buscado. O algoritmo utilizado nesta função será discutido com mais detalhes no Capítulo 4.

Outras quatro funções recebem trechos de textos em linguagem natural que são processados pelo analisador sintático antes de serem utilizados nas buscas na *ConceptNetBr*.

Elas permitem:

- Inferir um conceito a partir de sua descrição;
- Inferir os tópicos principais de um texto;
- Inferir o humor de um texto, classificando-o em um das seis categorias básicas de emoções: felicidade, tristeza, ira, medo, desgosto e surpresa;
- Sumarização de texto, através das sentenças consideradas mais importantes.

Vale ressaltar que essas quatro últimas funções ainda estão em desenvolvimento da versão brasileira da *API*.

#### **2.4.4.2. O Servidor XML-RPC**

Considerando todos os 156 mil fatos coletados até agora pelo site do projeto, a *ConceptNetBr* tem um tempo de carga de, aproximadamente, 15 segundos em uma máquina equipada com um processador Core 2 Duo e 2GB de memória RAM, ocupando cerca de 50MB na memória RAM depois de carregada, sem estar executando nenhuma de suas funções.

A fim de facilitar a distribuição de aplicações que utilizem as funções da *API* e tornar possível a utilização desses recursos por aplicações desenvolvidas em outras plataformas, como Java, e por máquinas com menor poder de processamento, foi desenvolvido um módulo servidor que roda sob o protocolo XML-RPC, baseado em um servidor mais simples do

projeto OMCS.

Devido ao formato utilizado para codificação de caracteres na biblioteca *Python* responsável pelo servidor XML-RPC, os textos que apresentavam acentos e caracteres especiais da língua portuguesa, como a cedilha, apresentavam erros de codificação quando transferidos pelo protocolo do servidor. Então, foi necessária a implementação de um mecanismo que encapsula o método que faz a codificação correta dos textos, solucionando o problema.

## 2.5. Considerações Finais

Neste capítulo foram apresentadas e discutidas algumas definições para o conceito de senso comum e a definição adotada neste trabalho. Foram apresentadas algumas iniciativas de prover o conhecimento de senso comum aos computadores pelos pesquisadores da inteligência artificial. Dentre eles está o projeto *Open Mind Common Sense*, projeto que serviu como base para o desenvolvimento da versão brasileira do projeto, *Open Mind Common Sense* no Brasil, contexto no qual está inserido este trabalho.

Foram apresentados os esforços do projeto brasileiro para a construção de uma base de conhecimento de senso comum em língua portuguesa. Desde o sítio desenvolvido e estratégias utilizadas para incentivar os colaboradores a fornecerem mais dados, até a arquitetura adaptada do projeto inicial e modificações necessárias para a construção da *ConceptNetBr*, permitindo a utilização do conhecimento de senso comum em língua portuguesa por aplicações computacionais.

Um grande diferencial do projeto brasileiro está na preservação do perfil dos colaboradores, o que permite às aplicações, que vão fazer uso deste conhecimento, levar em consideração o perfil de seu usuário e propósito da aplicação.

No contexto do mecanismo de analogias proposto neste trabalho, esse diferencial permite a criação de analogias considerando o conhecimento de senso comum do público alvo

para o qual o material instrucional está sendo desenvolvido. Considerar o conhecimento de senso comum durante a geração de analogias contribui para o sucesso do processo de aprendizagem, conforme poderá ser observado na discussão que se segue no próximo capítulo, onde é apresentado o relacionamento entre o senso comum, analogias e o processo de aprendizagem.

## **CAPÍTULO 3 - ANALOGIAS GERADAS COM O CONHECIMENTO DE SENSO COMUM E O PROCESSO DE APRENDIZAGEM**

---

### **3.1. Considerações iniciais**

Este capítulo discute a relevância da abordagem apresentada neste trabalho, onde analogias geradas com o uso de conhecimento de senso comum, que são parte do contexto dos estudantes, são geradas para serem utilizadas em materiais instrucionais. Para isso, faz-se necessária uma discussão da inter-relação existente entre esses três elementos: analogia, senso comum e processo de aprendizagem.

Neste capítulo é utilizada uma representação de conhecimento através de uma rede de nós interligada por predicados. Os nós representam os conceitos, aqui tratados como atômicos, e os predicados correspondem a preposições conectando conceitos. Os conhecimentos são escritos na forma: *predicado (conceito-1, conceito-2)*.

Este capítulo está organizado da seguinte forma. Na Seção 3.2 é discutido o conceito de analogia e apresenta-se uma classificação para comparações entre domínios proposta por Gentner(1983). Na Seção 3.3 mostra-se a relação entre senso comum, analogias e o processo de aprendizagem. Finalmente, são apresentadas algumas considerações finais referentes a este capítulo na Seção 3.4.

### **3.2. O Conceito de Analogia**

Quando as pessoas escutam uma analogia como “o fígado é como um filtro”, como elas entendem seu significado? Pode-se supor que elas comparam o conhecimento que elas têm sobre o fígado com o conhecimento que elas têm sobre filtros, e então tentem estabelecer relações entre esses conhecimentos. Segundo a Enciclopédia de Filosofia de Stanford (Ashworth, 2004), a palavra analogia vem do grego e significa proporcionalidade. Por muitas vezes foi traduzida para latim como *proportio* ou *proportionalitas*, se referindo a comparação entre duas proporções.

O Dicionário Houaiss da Língua Portuguesa (Houaiss, 2001) apresenta várias definições para a palavra analogia. A definição mais geral é a seguinte: “relação de semelhança entre coisas ou fatos”. Em seguida, o dicionário apresenta o conceito de analogia instanciado em vários contextos. No contexto da filosofia, por exemplo, analogia é definida como “identidade de relação entre pares de conceitos dessemelhantes, exemplificada pela proposição platônica: a inteligência está para a opinião assim como a ciência está para a crença”.

Filósofos gregos como Platão e Aristóteles utilizavam uma visão mais ampla do conceito de analogia (Shelley, 2003). Eles viam uma analogia como uma forma de abstração compartilhada. Para eles dois objetos análogos não compartilham somente relações, mas também idéias, padrões e atributos.

Pesquisadores da ciência cognitiva contemporânea vêm pesquisando sobre analogias utilizando uma visão mais próxima de Platão e Aristóteles. Desse estudo resultou a **Teoria de Mapeamento de Estruturas** (TME) de Gentner (1983).

Em sua teoria, Gentner explica uma comparação do tipo “*A é como B*” como sendo um mapeamento do domínio B para o domínio A. Onde um domínio é visto pela psicologia como um sistema de entidades, atributos e relações entre as entidades. O domínio A é chamado de **Alvo**, uma vez que é o domínio que está sendo explicado; e B é chamado de **Base**, uma vez que serve como fonte do conhecimento.

Gentner também faz uma distinção entre os diferentes tipos de comparações entre domínios, a saber, similaridade literal, analogia, e abstração, que são definidas a seguir:

### **Similaridade literal**

É uma comparação no qual um grande número de predicados é mapeado da base para o alvo em relação ao número de predicados não mapeados. Inclui mapeamento de relações (predicados) e de atributos (nós).

Por exemplo, a comparação “Marte é um planeta que, assim como a Terra, gira ao redor

do Sol, é menor que o Sol e é atraído pelo Sol” entre o domínio Terra e Marte é uma similaridade literal, pois tanto os predicados quanto os atributos de cada domínio são mapeados conforme se pode observar na Figura 13.

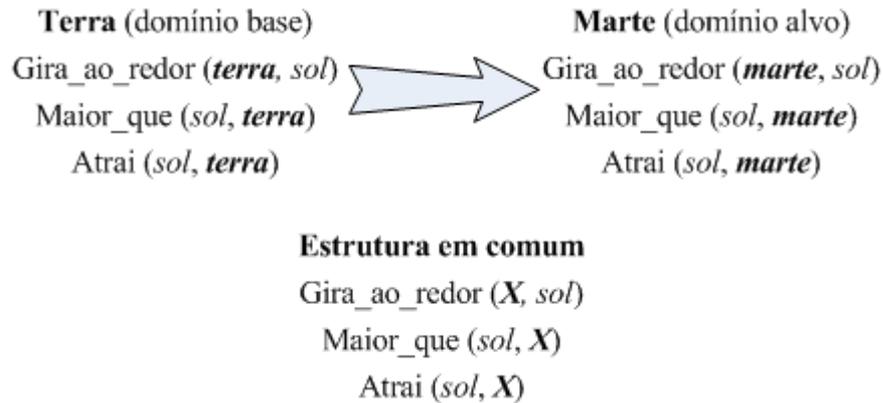


FIGURA 13. Exemplo de similaridade literal na comparação entre domínios.

### Analogia

É uma comparação onde há um grande número de mapeamentos entre as relações, porém com pouco ou nenhum mapeamento entre os atributos.

Por exemplo, a comparação, “O elétron gira ao redor do núcleo, é menor que o núcleo, e é atraído por ele, assim como o planeta Terra gira ao redor do Sol, é menor que o Sol, e é atraída por ele” é uma analogia, pois compartilha a estrutura de relações entre o domínio base e o domínio alvo, conforme pode ser observado na Figura 14.

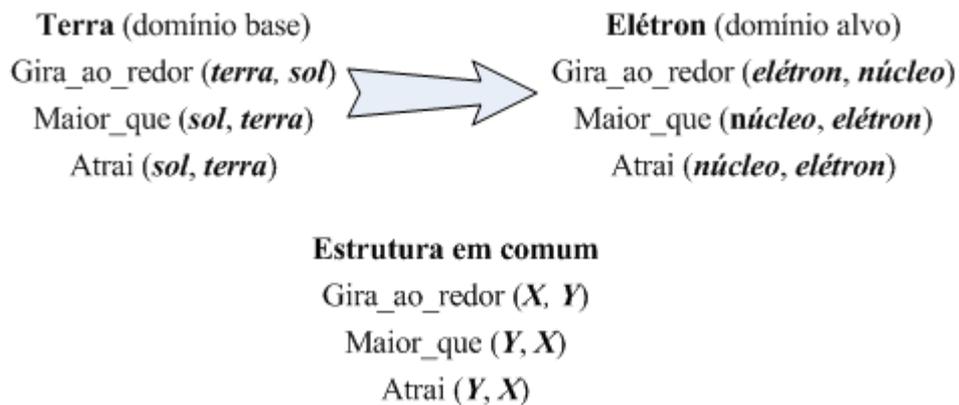


FIGURA 14. Exemplo de analogia na comparação entre domínios.

### Abstração

É uma comparação onde o domínio base é uma estrutura relacional abstrata. Por exemplo, a comparação “Marte é um planeta, que assim como os outros planetas, gira ao redor do Sol, é menor que o Sol, e é atraído pelo Sol” é considerada uma abstração, pois compartilha a estrutura de relações entre o domínio base e o domínio alvo, porém o domínio base é relacionado a uma estrutura abstrata, nesse exemplo “marte” pertence à classe de objetos nomeados “planetas”, conforme pode ser observado na Figura 15.

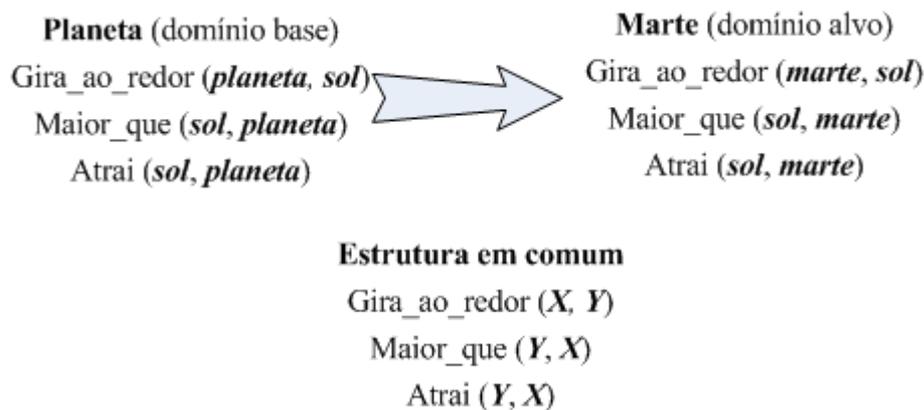


FIGURA 15. Exemplo de comparação entre domínios classificada como abstração.

Percebe-se que a relação entre similaridade literal e analogia não é de dicotomia, mas sim de extensão, uma vez que ambas compartilham o mapeamento de relações e se diferenciam pelo fato de que uma similaridade literal também considera mapeamento de atributos.

### 3.3. Potencialidade do Uso de Analogias com Senso Comum no Processo de Aprendizagem

O uso de analogias entre o novo conhecimento e o conhecimento prévio é uma estratégia comumente utilizada pelas pessoas quando elas querem entender algum novo conceito. E é freqüentemente utilizado para fazer outras pessoas entenderem um novo conceito que está sendo explicado. Isso ocorre por conta de que é mais fácil uma pessoa entender uma nova informação relacionando-a com sua própria experiência de vida (Freinet, 1993).

Nas teorias sobre o processo de aprendizagem, diversos autores mencionam a importância de se fazer uma relação entre o novo conhecimento, que está sendo ensinado, e o conhecimento prévio do aluno, considerando, desta maneira, o contexto cultural do aluno no processo de aprendizagem.

A abordagem utilizando analogias também pode ser justificada pela Teoria de Paulo Freire (Freire, 1996). Para Freire, os professores devem apresentar o novo conhecimento de uma maneira contextualizada. De acordo com esse autor, é inútil ensinar algo sem levar em consideração o contexto em que o aluno vive.

A idéia de Freire está em conformidade com a Teoria de Freinet. Para Freinet, se os professores vão ensinar para os alunos como contar, eles devem utilizar elementos do dia-a-dia dos alunos, isto é, se os alunos vivem em cidades, o professor deve utilizar-se de elementos como carros e lojas durante suas explicações. Se os alunos vivem em áreas rurais, o professor deve utilizar-se de elementos como animais e plantas, para fazer com que os alunos entendam o que está sendo explicado.

Ausubel(1976) explica essa questão em sua teoria quando define a Aprendizagem Significativa. Segundo Ausubel, Aprendizagem Significativa é o processo pelo qual uma nova informação relaciona-se a um aspecto relevante da estrutura de conhecimento do indivíduo, chamado de **conceito subsunçor** ou somente **subsunçor**.

A Aprendizagem Significativa ocorre quando o novo conhecimento ancora-se em subsunçores na estrutura cognitiva do aprendiz, implicando o crescimento e modificação do conceito subsunçor. Ou seja, para que possa ocorrer uma aprendizagem significativa, o novo conhecimento, que está sendo ensinado, deve ser apresentado ao aluno de forma a fazer o aluno relacionar o novo conhecimento com outros conhecimentos que ele tem em sua estrutura cognitiva.

Na Figura 16 ilustra-se o processo cognitivo subjacente da Aprendizagem Significativa.

Na parte 1 se vê a estrutura cognitiva do aluno representada por círculos ligados a outros círculos, os subsunçores representados pelos círculos cinza e o novo conhecimento representado pelo círculo preto. Com a chegada do novo conhecimento, na Parte 2 se vê que dois subsunçores são ativados, e o novo conhecimento é ancorado nesses subsunçores, passando assim a fazer parte da estrutura cognitiva do aluno.

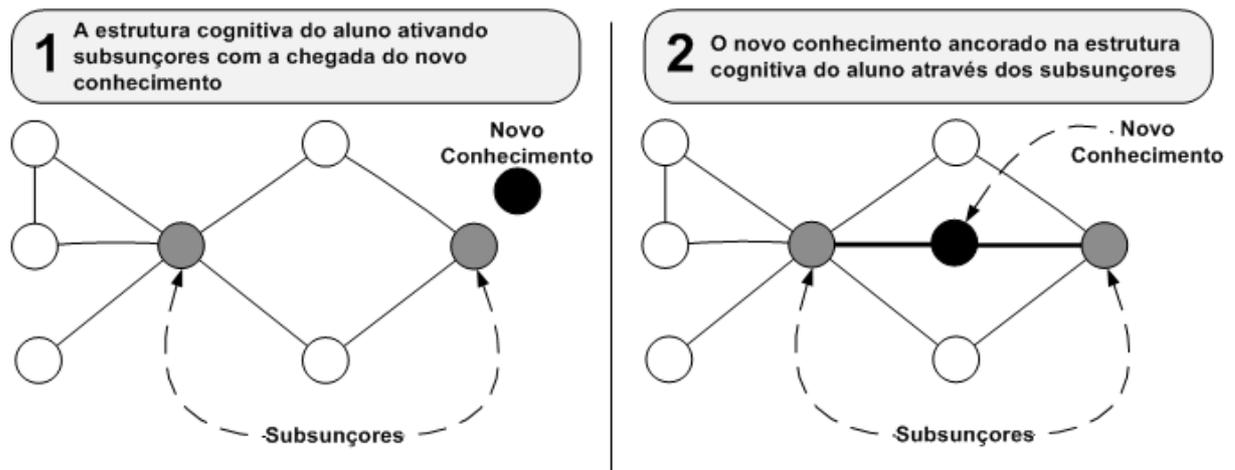


FIGURA 16. Alteração da estrutura cognitiva do aluno pela ocorrência da aprendizagem significativa.

Pode-se verificar a conformidade entre a Teoria da Aprendizagem Significativa de Ausubel o uso de analogias no processo de aprendizagem, considerando o domínio base da analogia representando os subsunçores e o novo conhecimento representando o domínio alvo.

Observa-se um exemplo dessa conformidade na Figura 17, onde o professor está explicando o que é uma lichia para seus alunos através de uma analogia do tipo: “*A lichia é uma fruta assim como uma laranja lima, e a lichia tem sabor doce assim como uma laranja lima*”. Neste ponto, o professor parte do pressuposto que os fatos “*Laranja lima é uma fruta*” e “*Laranja lima é doce*” são de conhecimento dos alunos. Ou seja, o professor supõe que esses conhecimentos pertençam ao conhecimento de senso comum dos alunos, que por sua vez, ativam seus subsunçores “*fruta*” e “*doce*”. Então, o novo conhecimento “*lichia*” é conectado à estrutura cognitiva do aluno por meio de relações de mesmo tipo à laranja lima para seus subsunçores “*fruta*” e “*doce*”, conforme ilustrado na parte 2 da Figura 18.

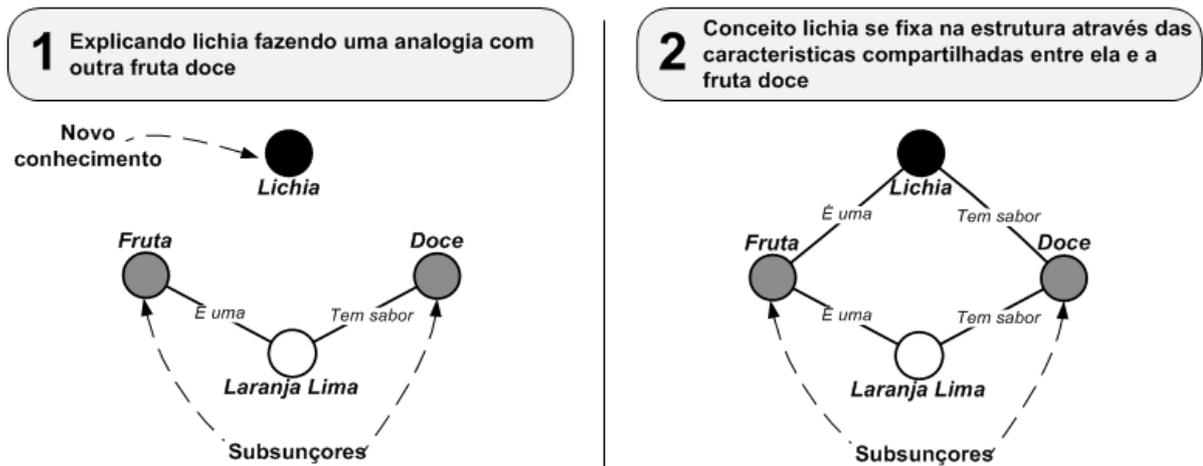


FIGURA 17. Exemplificando a conformidade do uso de analogias e o processo de Aprendizagem Significativa.

Essa comparação presente no exemplo supracitado é classificada pela teoria de Gentner como uma similaridade literal, pois compartilha os argumentos “Fruta” e “Doce” e os tipos das relações “é um” e “tem sabor”.

A Teoria Instrucional de Gagné (1974) aborda os processos internos de aprendizagem através de itens que foram denominados domínios. Um desses domínios é constituído pelas **estratégias cognitivas**, que segundo ele são capacidades internamente organizadas que o aluno usa para guiar seus próprios processos de atenção, aprendizagem, memória e pensamento. O aluno usa uma estratégia cognitiva, por exemplo, ao prestar atenção nas diversas características daquilo que está lendo. O leitor usa certas estratégias cognitivas para selecionar e codificar o que aprende, utilizando outras estratégias para recuperar posteriormente essas informações. Considerando essa definição, estratégias cognitivas são os meios de que o aluno dispõe para administrar seu processo interno de aprendizagem.

Considerando a Teoria Instrucional de Gagné, alguns autores, como Beckman (2002) e West *et al.* (1991), têm trabalhado na identificação de estratégias que os alunos comumente usam em seu processo de aprendizagem.

Nesse contexto, outros pesquisadores, como Liebman (1998) e Neris (2005), têm investigado como os professores podem utilizar agentes que possam estimular os alunos a

usarem suas próprias estratégias cognitivas: esses agentes são chamados **operadores cognitivos**. Neris mostrou que a inserção de operadores cognitivos no material instrucional aumenta a usabilidade do hiperdocumento que contém tal material. Ela formalizou as estratégias cognitivas adotadas com sucesso no ensino presencial por Liebman (1998).

No grupo de estratégias identificadas por Beckman, está uma estratégia chamada “Associando”. Essa estratégia pode ser diretamente associada a analogias, uma vez que analogias fazem um mapeamento entre dois domínios. Além disso, analogias são frequentemente utilizadas para ajudar na explicação de um conceito como uma associação entre o novo conceito e o conhecimento prévio presente na estrutura cognitiva do aluno, fato que motivou o desenvolvimento do segundo protótipo que será apresentado na Seção 5.3.2.

Além disso, West *et al.* (1991), também considera que “Metáforas e Analogias” podem ser consideradas um operador cognitivo. Por exemplo, o professor pode apresentar alguns textos na forma de analogias aos alunos, para que eles possam ativar a habilidade interna de analogias por si próprios.

Stamm (2003) menciona que para manter os alunos envolvidos no processo de aprendizagem, o professor deve apresentar uma história para os alunos e depois perguntar como aquela história se relaciona com a matéria, com o objetivo de incentivar a ativação da estratégia cognitiva “Analogias” dentro da mente dos alunos.

### **3.4. Considerações Finais**

Este capítulo mostrou a relação entre analogias, senso comum e o processo de aprendizagem. É interessante observar como as teorias sobre o processo de aprendizagem mencionadas estão em conformidade umas com as outras em relação às analogias e a importância de levar em consideração o conhecimento de senso comum.

Resumidamente, as teorias de aprendizagem mencionadas estão relacionadas com analogias e com senso comum pelo seguinte raciocínio: o uso do operador cognitivo

“Metáforas e Analogias” através do uso de analogias entre o conhecimento de senso comum e o novo conhecimento estimula a associação entre o novo conhecimento e conhecimentos que já estavam na mente do aluno, isto é, na estrutura cognitiva, através dos subsunçores. Fazer essas conexões significa promover a Aprendizagem Significativa e isso somente é possível porque o novo conhecimento está relacionado com algo que o aluno já sabe, ou seja, ao conhecimento prévio do aluno.

Independentemente da classificação proposta por Gentner, este trabalho considera todos os tipos de relações entre domínios úteis para o processo de aprendizagem, desde que, todos eles tenham o domínio base da comparação fundamentados no conhecimento de senso comum dos alunos. Pois, assim, os conhecimentos do domínio base podem atuar como subsunçores, ancorando o novo conhecimento na estrutura cognitiva do aluno e promovendo assim a Aprendizagem Significativa.

No próximo capítulo é apresentado o algoritmo desenvolvido durante este trabalho de mestrado para a geração de analogias considerando o formato da base de conhecimento de senso comum do Projeto *Open Mind Common Sense* no Brasil, bem como o algoritmo que serviu de base para sua implementação.

## **CAPÍTULO 4 - GERANDO ANALOGIAS ATRAVÉS DA BASE DE CONHECIMENTO DO PROJETO OMCS-BR**

---

### **4.1. Considerações Iniciais**

Este capítulo apresenta o algoritmo para geração de analogias desenvolvido neste trabalho, bem como o algoritmo que serviu de base para sua implementação. Serão ressaltadas as mudanças realizadas devido às especificidades do formato utilizado para representação de conhecimento no projeto OMCS-Br. No intuito de auxiliar a explicação do algoritmo, alguns pontos da Teoria de Mapeamento de Estruturas - TME também serão retomados e alguns detalhes serão mais bem elucidados.

Para isso, este capítulo está organizado da seguinte maneira. Primeiramente, na Seção 4.2 retoma-se a discussão sobre a TME, apresentando-se maiores detalhes que poderão facilitar o entendimento do algoritmo. A Seção 4.3 apresenta o algoritmo chamado Mecanismo de Mapeamento de Estruturas - MME que foi utilizado como base para implementação do algoritmo desenvolvido neste trabalho. A Seção 4.4 descreve o algoritmo desenvolvido e as modificações realizadas. Algumas considerações finais são apresentadas na Seção 4.5.

### **4.2. A Teoria de Mapeamento de Estruturas - TME**

Segundo a teoria de Mapeamento de Estruturas, mencionada na Seção 3.2, uma analogia é um mapeamento do conhecimento de um domínio base em outro domínio alvo. Nesse mapeamento, define-se que um conjunto de relações existente no domínio base deve também existir no domínio alvo.

Um atributo do domínio base não precisa se assemelhar ao seu correspondente no domínio alvo, porém a estrutura de relações deve ser mantida. Essa correspondência entre os atributos é feita através das posições que os atributos ocupam na estrutura relacional em comum.

Dadas duas coleções de objetos  $\{b_1\}$  e  $\{a_1\}$ , representando os domínios base e alvo

respectivamente, esse mapeamento  $M$  é representado pela seguinte:

$$M: b_1 \rightarrow a_1$$

Os predicados são mapeados segundo os seguintes tipos de mapeamentos:

- Relações entre objetos na base são mapeadas no alvo. Por exemplo:

$$\text{MaiorQue}(b_i, b_j) \rightarrow \text{MaiorQue}(a_i, a_j)$$

- Relações de alta ordem são mapeadas se mantiveram a mesma estrutura de predicados na base e no alvo. Por exemplo:

$$\text{Conseqüência}(\text{Arremessar}(b_1, b_j), \text{Quebrar}(b_j, b_k)) \rightarrow \\ \text{Conseqüência}(\text{Arremessar}(a_1, a_j), \text{Quebrar}(a_j, a_k))$$

- Atributos dos objetos na base e no alvo não são mapeados caso a comparação seja do tipo analogia. Por exemplo:

$$\text{Amarelo}(b_1) \rightarrow \text{Amarelo}(a_1)$$

O termo “relações de alta ordem” não tem ligação com os termos “lógica de primeira ordem” e “lógica de segunda ordem”, comumente utilizados pela IA. A ordem de um predicado é calculada, na TME, da seguinte maneira: entidades têm ordem 0; a ordem de um predicado é calculada como sendo 1 mais a máxima ordem de seus argumentos.

### 4.3. O Mecanismo de Mapeamento de Estruturas - MME

O Mecanismo de Mapeamento de Estruturas – MME (Falkenhainer *et al.*, 1989), foi desenvolvido para explorar os aspectos computacionais da TME. Esse mecanismo é flexível, permitindo testar os diferentes tipos de comparações definidos na teoria de Gentner, bastando,

para isso, alterar a coleção de regras de casamentos.

As regras de casamentos guiam a construção da analogia e especificam quais fatos e entidades na base e no alvo são possíveis mapeamentos e atribui um peso ao possível mapeamento representando o grau de confiança. Diferentes regras de mapeamentos geram diferentes tipos de comparações. Assim, essas regras são a chave da flexibilidade do algoritmo.

A analogia é processada em três passos que serão apresentados a seguida. No primeiro passo, apresentado na Seção 4.3.1, todos os potenciais pares entre base e alvo, chamados de **mapeamentos locais**, são construídos e avaliados pelas regras de mapeamento. No segundo passo, apresentado na Seção 4.3.2, todas as combinações consistentes dos casamentos locais são geradas para formar os **mapeamentos globais**. Na seção 4.3.3 é apresentada o terceiro passo, no qual os mapeamentos globais são avaliados e recebem uma pontuação.

#### 4.3.1. Construção dos Mapeamentos Locais

As regras de casamento são aplicadas para todas as combinações possíveis de entidades e predicados da base com entidades e predicados do alvo. Essas regras são compostas de uma condição e um corpo. Se a condição for satisfeita, o corpo cria o mapeamento local correspondente.

Um exemplo, de regra de casamento poderia ter a condição: “*se os dois predicados são iguais*”. Então, caso a condição seja satisfeita, o corpo da regra cria um casamento local mapeando os argumentos. Após a criação de todos os casamentos locais, um conjunto de regras de evidências dá um peso a cada casamento, indicando-lhe um grau de confiança.

Durante a explicação desse algoritmo, um mapeamento local ligando o item da base com o item do alvo será denotado por  $MH(i)$ , no intuito de manter a mesma nomenclatura utilizada pelo autor do algoritmo.

No final, tem-se uma lista de pares com seus respectivos pesos. Cada par é composto por um predicado ou entidade do alvo e um predicado ou entidade da base, que serão combinados no passo 2 do algoritmo para a geração dos mapeamentos globais.

#### 4.3.2. Construção dos mapeamentos globais

Um casamento global, chamado de *gmap* em consonância com a nomenclatura utilizada pelo autor do algoritmo, é um conjunto consistente de casamentos locais. Para ser considerado consistente, um *gmap* deve obedecer ao princípio um-para-um. Ou seja, cada item do domínio base deve estar mapeado a um, e somente um, item do domínio alvo, assim como cada item do domínio alvo também deve estar mapeado a um, e somente um, item do domínio base.

A verificação da consistência é o principal problema encontrado na construção dos casamentos globais. Para isto, para cada mapeamento local é computado o seguinte: (a) o mapeamento de entidades que ele engloba; (b) o conjunto de mapeamentos locais que é inconsistente com ele.

Para obter o mapeamento de entidades englobadas por um mapeamento local  $MH(b_i, a_j)$ , envolvendo o item da base  $b_i$  e o item do alvo  $a_j$ , é realizado o seguinte procedimento: Se  $b_i$  e  $a_j$  são expressões, então elas devem estar em qualquer coleção que tenha  $MH(b_i, a_j)$ . Aplicando-se essa restrição recursivamente, todos os descendentes de  $MH(b_i, a_j)$  devem estar na mesma coleção. Uma vez que qualquer cadeia de descendentes finaliza com pares de hipóteses envolvendo entidades, cada par implica um conjunto específico de entidades correspondentes.

Um mapeamento de entidades englobadas por  $MH(b_i, a_j)$  é chamado de *emap*.  $Emap(MH(b_i, a_j))$  é simplesmente a união dos *emaps* suportados pelos descendentes de  $MH(b_i, a_j)$ .

Para garantir o mapeamento um-para-um definido pela TME, é associado a cada  $MH(b_i, a_j)$  um conjunto de casamentos que fornecem mapeamentos alternativos para  $b_i$  e  $a_j$ .

Portanto, nenhum membro deste grupo pode estar no mesmo grupo que  $MH(b_i, a_j)$ .

Data uma hipótese de casamento  $MH(b_i, a_j)$ , o conjunto  $Conflitantes(MH(b_i, a_j))$  consiste nos conjuntos de hipóteses de casamentos que apresentam mapeamentos alternativos para para  $b_i$  e  $a_j$ . Esse conjunto pode ser formalmente definido da seguinte forma:

$$Conflitantes(MH(b_i, a_j)) \equiv \left[ \bigcup_{b_k \in base} \{MH(b_k, a_j) \mid b_k \neq b_i\} \right] \cup \left[ \bigcup_{a_k \in alvo} \{MH(b_i, a_k) \mid a_k \neq a_j\} \right]$$

O conjunto  $Conflitantes(MH(b_i, a_j))$  somente representa as inconsistências locais. Entretanto, pode-se usar o conjunto  $Conflitantes$  em conjunto com  $Emap(MH(b_i, a_j))$  para definir recursivamente o conjunto de todas as hipóteses de casamento que nunca podem estar no mesmo Gmap que  $MH(b_i, a_j)$ .

Sendo  $MH_i$  um mapeamento local ou global, o conjunto  $NoGood(MH_i)$  é o conjunto de todas as hipóteses de casamento que nunca podem estar no mesmo grupo que  $MH_i$ . O conjunto é recursivamente definido da seguinte forma: Se  $MH_i$  é um mapeamento local, então  $NoGood(MH_i) = Conflitantes(MH_i)$ . Caso contrário,  $NoGood(MH_i)$  é a união dos conjuntos conflitantes de todos os seus descendentes, a saber:

$$NoGood(MH_i) = Conflitantes(MH_i) \cup \bigcup_{MH_j \in Argz(MH_i)} NoGood(MH_j)$$

Os conjuntos  $Conflitantes$ ,  $Emap$  e  $NoGood$  são calculados na seguinte ordem. Em primeiro lugar, o conjunto conflitantes é calculado para cada hipótese de casamento, uma vez que ele requer somente informações locais. Em seguida, os conjuntos  $Emaps$  e  $NoGood$  são

calculados para cada *Emap*. Por último, *Emaps* e *NoGood* são calculados para todos os mapeamentos locais pela propagação dos resultados dos *Emaps* em seus antecessores.

Os *gmaps* gerados pelo algoritmo devem ser consistentes e máximos. Para um *gmap* ser considerado consistente, a interseção do conjunto *Emaps* com o conjunto *Conflitantes* deve ser nula, ou seja, o seguinte deve ocorrer:

$$\text{Consistente}(GMap) \Leftrightarrow \text{Emaps}(GMap) \cap \text{Conflitantes}(GMap) = \emptyset$$

Um *gmap* é considerado máximo quando qualquer outro *gmap* que for unido a ele o torne inconsistente.

Tendo esses conjuntos calculados, tem início o processo de combinação dos mapeamentos locais em mapeamentos globais. Os mapeamentos locais de maior ordem são utilizados como *gmaps* iniciais, pois, caso contrário, todos os mapeamentos e seus descendentes deveriam ser considerados como iniciais, aumentando o custo computacional.

A combinação dos *gmaps* iniciais é realizada através de uniões entre eles. O *gmap* resultante deve ser consistente. A consistência da união entre o *Gmap<sub>i</sub>* e o *Gmap<sub>j</sub>* é calculada pela fórmula:

$$\begin{aligned} \text{Consistente}(Gmap_i, Gmap_j) \leftrightarrow & \text{Elementos}(Gmap_i) \cap \text{NoGood}(Gmap_j) = \emptyset \\ & \wedge \text{Elementos}(Gmap_j) \cap \text{NoGood}(Gmap_i) = \emptyset \end{aligned}$$

Primeiramente, são combinados os *gmaps* que têm algum mapeamento em comum e, finalizando, é realizada a combinação dos *gmaps* sem mapeamento em comum.

Assim são obtidos mapeamentos globais máximos e consistentes, que serão avaliados durante o último passo do algoritmo.

### 4.3.3. Avaliação dos Mapeamentos Globais

A avaliação dos mapeamentos globais gerados pelo segundo passo do algoritmo é realizada através da soma dos pesos de cada um dos mapeamentos locais que o compõem. Outros métodos chegaram a ser considerados, porém, segundo Falkenheiner *et al.*(1990) isso limitaria a flexibilidade do algoritmo, que está baseada nas regras de casamento e evidências do primeiro passo. Assim, os pesos podem ser ajustados alterando as regras de evidências de acordo com a instanciação do algoritmo realizada.

## 4.4. Implementação de um algoritmo baseado no MME considerando a base OMCS-Br

Várias adaptações foram necessárias para que fosse possível a implementação de um algoritmo baseado no MME utilizando a base de conhecimento do projeto OMCS-Br. Essas adaptações foram devido às diferenças existentes entre o formato das bases de conhecimentos utilizadas pelo MME e o formato da base de conhecimento de senso comum do projeto OMCS-Br. Com a finalidade de melhor explicar e justificar as decisões tomadas durante a implementação do algoritmo, serão ressaltadas três diferenças importantes existentes entre o formato das duas bases, antes da apresentação da implementação desenvolvida.

Primeiramente, conforme apresentado na Seção 2.4.3, a base do projeto OMCS-Br, seguindo o projeto original, é composta por relações binárias entre dois conceitos que utilizam um conjunto fixo composto por 20 diferentes predicados. Já a base utilizada pelo MME não tem limitações em relação aos tipos de predicados. Esses podem representar desde um relacionamento ‘*comum*’ como, por exemplo, o relacionamento do tipo *é-um*, até mesmo uma relação com o nome de *sinhal-oposto* para descrever a diferença de carga entre um próton e um elétron.

Em segundo lugar, também se pode observar um diferença na representação dos

atributos. Na base do projeto OMCS-Br, um atributo de um objeto é representado na forma “*PropertyOf*(‘carro’ ‘bonito’)”, enquanto na base utilizada pelo MME um atributo é representado na forma ‘*bonito(carro)*’. Essa identificação do predicado *PropertyOf* como modo de representação de atributos adotado pelo projeto OMCS-Br é relevante, na medida em que permite a sua exclusão, que é prevista pela TME caso o mapeamento seja do tipo analogia.

Em terceiro lugar, a base utilizada pelo MME permite que predicados sejam utilizados como argumentos de outros predicados, chamados no contexto do MME de predicados de *alta ordem*. O mesmo não é observado na base do projeto OMCS-Br, onde os predicados são compostos sempre por dois argumentos representando elementos da rede de conceitos.

A implementação realizada neste trabalho segue a mesma divisão em três passos utilizados pelo MME: construção dos mapeamentos locais; construção dos mapeamentos globais e; avaliação dos mapeamentos globais. A seguir cada passo será explicado detalhadamente nas Seções 4.4.1 a 4.4.3, respectivamente.

#### **4.4.1. Construção dos Mapeamentos Locais**

O primeiro passo para a construção dos mapeamentos locais é a seleção das relações da base que serão utilizadas para a construção dos mapeamentos.

Considerando que neste trabalho considera-se, conforme mencionado no final do Capítulo 3, todos os tipos de relações entre domínios úteis para o processo de aprendizagem, todos os predicados do tipo *PropertyOf* podem ser tratados de duas formas: 1) são retirados da lista das relações utilizadas pelo algoritmo ou; 2) são mantidos, porém têm seu peso reduzido.

A retirada ou diminuição do peso das relações do tipo *PropertyOf*, é passado como parâmetro durante a execução do algoritmo e não significa que comparações por similaridades literais e abstração não serão criadas, pois comparações por similaridades literais são um subconjunto das analogias e comparações por abstração também são geradas, considerando

que a base de senso comum utilizada apresenta informações sobre diferentes domínios em diferentes níveis de abstrações.

Tendo os conjuntos base e alvo definidos, para cada possível combinação de relações, uma da base e uma do alvo são aplicadas as seguintes regras de casamento:

- Se o predicado das duas relações é o mesmo, mas os dois atributos são diferentes, então é criado um mapeamento na forma:

$$\mathbf{PredicadoX}(b_i, b_j, f_b, i_b) \wedge \mathbf{PredicadoX}(a_i, a_j, f_a, i_a) \rightarrow \\ \mathbf{PredicadoX}((b_i, a_i), (b_j, a_j), (f_b + f_a + (i_b + i_a)/2) * 0.2)$$

- Se o predicado e um dos argumentos das duas relações são o mesmo, e o argumento igual é um sintagma verbal, então é criado um mapeamento na seguinte forma:

$$\mathbf{PredicadoX}(V, b_j, f_b, i_b) \wedge \mathbf{PredicadoX}(V, a_j, f_a, i_a) \rightarrow \\ \mathbf{PredicadoX}((V, V), (b_j, a_j), (f_b + f_a + (i_b + i_a)/2) * 0.5)$$

Ou, caso sintagma verbal esteja no segundo argumento, na seguinte forma:

$$\mathbf{PredicadoX}(b_i, V, f_b, i_b) \wedge \mathbf{PredicadoX}(a_i, V, f_a, i_a) \rightarrow \\ \mathbf{PredicadoX}((b_i, a_i), (V, V), (f_b + f_a + (i_b + i_a)/2) * 0.5)$$

A restrição quanto ao tipo do argumento ser um sintagma verbal está relacionada com a diferença de representação de sintagmas verbais nas duas bases. Na base utilizada pelo MME, todos os verbos são mapeados como predicados. Contudo, a base do projeto OMCS-Br, por ter uma lista fixa de predicados, utilizam-se nós da rede para também representar sintagmas

verbais. Assim adotou-se essa abordagem no intuito de gerar mapeamentos locais mais próximos dos gerados pelo MME.

Considera-se a solução adotada uma solução híbrida, que apresenta características de duas outras abordagens possíveis:

- **O predicado e um dos atributos devem ser iguais:** Porém o algoritmo geraria somente comparações por similaridade literal.
- **Somente o predicado deve ser igual:** Porém aumentaria muito o número de mapeamentos locais, aumentando muito o tempo de processamento. Em vários testes realizados com redes semânticas com cerca de nove mil nós cada, a execução do algoritmo consumiu mais de seis gigabytes de memória, sendo dois gigabytes de memória RAM e quatro gigabytes de memória virtual e travou o computador após várias horas de execução.

O cálculo do peso de cada mapeamento local segue a mesma abordagem utilizada pela *API ConceptNet*, onde o peso total de uma relação é calculada pela soma do parâmetro frequência  $f$  com a metade do parâmetro inferência  $i$ .

No final deste passo se tem uma nova rede semântica, onde cada nó é composto por uma tupla contendo um nó da base e outro do alvo. Essa construção somente foi possível devido a todas as regras de casamento terem uma característica em comum: os predicados das relações devem ser iguais. A seguir será descrito como essa rede semântica será utilizada para a geração dos mapeamentos globais.

#### 4.4.2. Construção dos Mapeamentos Globais

O MME utiliza os mapeamentos com predicados de ordem mais elevadas para calcular os mapeamentos globais iniciais. Na base do projeto OMCS-Br, por sua vez, todos os predicados são de ordem 1 segundo a classificação do MME, o que impossibilita a adoção da

mesma estratégia para a criação dos mapeamentos globais.

Uma possibilidade seria a simples combinação de todos os possíveis mapeamentos locais, porém levaria a um crescimento exponencial de possibilidades, inviabilizando o seu uso.

A solução encontrada utiliza o termo para o qual se quer uma analogia, que é passado como parâmetro para o algoritmo, para a criação dos mapeamentos globais, através de uma adaptação do algoritmo *spread activation* (Preece, 1981), aplicado na nova rede semântica gerada.

Seja  $X$  o termo para o qual se deseja uma analogia e cada nó da nova rede semântica composto por dois objetos na forma  $(base, alvo)$ , um mapeamento global será construído para todos os nós que tenham  $X$  como elemento do alvo  $(base, X)$ . Assim, todos os elementos da base que forem mapeados com  $X$  são possíveis analogias de  $X$ .

A construção de cada um dos mapeamentos globais é realizada utilizando uma variante do algoritmo de *spread activation*. Na variante desenvolvida neste trabalho, o algoritmo, antes de ativar um nó da rede, verifica a compatibilidade do novo nó com os nós já ativados, substituindo, assim, o cálculo do conjunto *Conflitantes* presente no segundo passo do algoritmo MME original.

A verificação da compatibilidade de um nó antes de ser ativado, possibilita garantir que o mapeamento global obedecerá ao princípio um-para-um definido pela TME, onde um item da base tem que estar mapeado a um, e somente um, item do alvo e cada elemento do alvo tem que estar mapeado a um, e somente um, item da base.

#### **4.4.3. Avaliação dos Mapeamentos Globais**

Tendo os mapeamentos globais construídos, é necessária uma avaliação para saber qual ou quais devem ser as saídas finais do algoritmo. O MME faz essa avaliação através da simples soma dos pesos dos mapeamentos globais que compõem o mapeamento global,

visando uma maior flexibilidade do algoritmo, que foi inicialmente desenvolvido para explorar os aspectos computacionais da TME.

Em uma primeira versão do algoritmo desenvolvido neste trabalho, essa mesma abordagem foi adotada. Porém, verificou-se empiricamente a geração de muitas analogias compostas por vários mapeamentos de um mesmo predicado sendo avaliadas com peso mais alto do que analogias contendo um número menor de mapeamentos, porém envolvendo diferentes tipos de predicados. Considera-se que a existência de diferentes predicados mapeando os mesmos itens na base e no alvo é um forte indicativo para um mapeamento, considerando que a TME está fundamentada no compartilhamento da estrutura relacional.

Assim uma nova versão foi implementada, levando em consideração o número de predicados diferentes presentes do mapeamento global durante a avaliação. A solução adotada neste projeto foi multiplicar o peso do mapeamento global anteriormente obtido pelo número de diferentes predicados que ele utiliza.

O algoritmo procura otimizar a realização das tarefas de verificação de conflitos em um mapeamento para tornar viável a sua utilização em aplicações computacionais que exigem um baixo tempo de resposta. Mesmo assim, durante os experimentos realizados somente foi utilizado um subconjunto das relações da base do projeto OMCS-Br classificadas como concisas, descartando-se as relações do tipo K-line. No projeto OMCS, uma relação é denominada concisa se a sua frequência ou inferência é maior que 1. A utilização da base completa ainda resulta em problemas de alto consumo de memória e alto tempo de processamento.

#### **4.5. Considerações Finais**

Neste capítulo apresentou-se o algoritmo desenvolvido neste trabalho para geração de analogias utilizando a base de conhecimento de senso comum, bem como o algoritmo MME, utilizado como base para sua implementação. Procurou-se ressaltar as diferenças entre as

bases utilizadas pelos dois algoritmos a fim de justificar as modificações realizadas.

Em relação à avaliação dos mapeamentos gerados, um fator que contribui para a geração de mapeamentos contendo somente um tipo de predicado é a distribuição irregular no número de relações de cada predicado presente na base do projeto OMCS-Br. Pois, durante a criação do projeto, os gabaritos foram colocados gradualmente no sítio, além do fato da colaboração ser voluntária; assim, o colaborador escolhe em qual gabarito quer colaborar.

Um exemplo dessa distribuição irregular é o predicado *LocationOf*, cujo gabarito está no sítio desde sua primeira versão, que representa cerca de 21% das relações presentes na *ConceptNetBr*, desconsiderando-se as relações do tipo K-line. Porém, com o crescimento da base através de todos os gabaritos disponíveis no sítio, a tendência é a atenuação dessa diferença.

Ainda não se conseguiu estudar a influência da presença das relações do tipo *PropertyOf* nas analogias geradas, pois o gabarito que captura esta relação foi disponibilizado no sítio do projeto OMCS-Br há pouco tempo, havendo um número relativamente pequeno de relações com esse predicado.

O próximo capítulo apresenta a utilização do algoritmo desenvolvido em dois *softwares* com propósitos educacionais: um sistema de apoio a treinamento a distância e um editor de material instrucional baseado em padrões.

## **CAPÍTULO 5 - APLICANDO A GERAÇÃO DE ANALOGIAS EM APLICAÇÕES EDUCACIONAIS**

---

### **5.1. Considerações iniciais**

Este capítulo apresentam-se dois protótipos, o segundo ainda em desenvolvimento, que utilizam o algoritmo apresentado no capítulo anterior. Esses protótipos foram construídos no intuito de mostrar a potencialidade do algoritmo desenvolvido neste trabalho em aplicações educacionais, validando a abordagem adotada.

O primeiro protótipo, apresentado na Seção 5.2, é um sistema de apoio para treinamento a distância. Ele foi desenvolvido visando a uma parceria entre o LIA e uma empresa que presta serviços de treinamentos à distância para grandes companhias. Dentre os treinamentos realizados por essa empresa, está um treinamento relacionado à segurança no trabalho, fato que motivou o desenvolvimento desse protótipo.

O segundo protótipo, apresentado na Seção 5.3, desenvolvido dentro do contexto de outro projeto desenvolvido pelo LIA chamado Cognitor (Talarico Neto, 2006). Este projeto visa à construção de um editor de material instrucional apoiado na Linguagem de Padrões - LP Cog-Learn (Talarico Neto, 2005). Os padrões da linguagem Cog-Learn fornecem soluções de sucesso para problemas recorrentes encontrados pelos professores durante o projeto e edição de material instrucional para ambientes WEB de ensino a distância – EAD. Nessa linguagem, um dos padrões está relacionado com inserção de analogias no material instrucional, fato que motivou a implementação de um mecanismo que auxilie o professor nessa tarefa. Apesar de não finalizado, optou-se pela inclusão desse protótipo neste trabalho, por demonstrar a potencialidade do mecanismo desenvolvido para ser utilizado em outros contextos educacionais.

Finalizando este capítulo, a Seção 5.4 apresenta algumas considerações finais sobre os protótipos desenvolvidos.

## **5.2. Um Sistema de Apoio para Treinamento a Distância.**

Considerando-se as restrições de tempo e espaço que as pessoas encontram durante sua jornada de trabalho, os treinamentos a distância realizados por computador podem ser uma solução para conciliar essas restrições com a necessidade de uma educação contínua exigida pelo mercado (Zhang, 2004).

No entanto a distância física entre os professores e alunos, que nesse contexto são representados pelos empregados da empresa, reduz as oportunidades do professor identificar as dificuldades dos alunos e intervir no processo de ensino a tempo de corrigir conceitos mal interpretados e fornecer mais informações ao aluno que necessitar (Doube, 2000). Mais especificamente, a clareza do material apresentado ao aluno é um fator determinante, uma vez que esse material tem a capacidade de esclarecer questões que venham a aparecer durante seus estudos.

Surge assim a necessidade de um sistema que apóie o aluno quando o professor não está presente, ajudando no andamento dos estudos do aluno, até a próxima oportunidade de interação direta com o professor. Com a finalidade de suprir tais anseios foi desenvolvido o protótipo apresentado a seguir nas Subseções 5.2.1 e 5.2.2.

### **5.2.1. A Implementação do Sistema de Apoio**

Esse sistema é disponibilizado em um site, no qual os alunos podem acessar a qualquer momento durante seus estudos para obter maiores informações sobre um determinado assunto ou termo. Uma das principais características do sistema de apoio desenvolvido é a utilização de analogias para explicar o termo pesquisado. Porém, para a construção de analogias entre o conhecimento de senso comum e o conhecimento sobre segurança do trabalho é necessário ter uma base contendo esse conhecimento especialista. Além disso, essa base deve estar no formato utilizado pelo algoritmo desenvolvido neste trabalho.

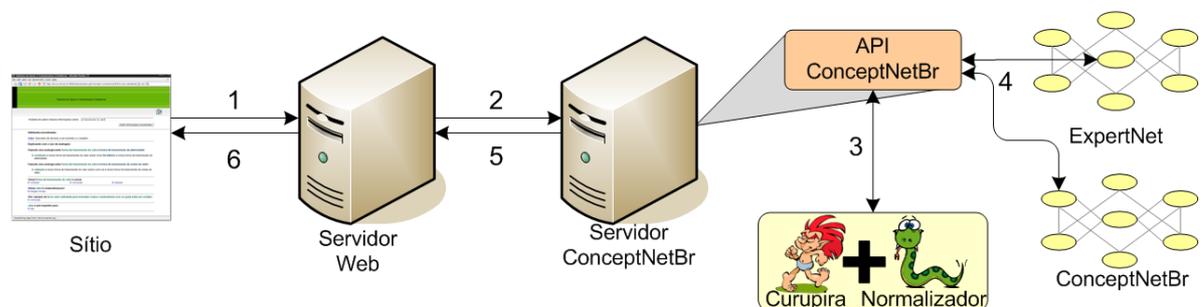
A base foi construída manualmente pelo autor deste trabalho, que atuou como

engenheiro do conhecimento. Foi utilizado como base um material instrucional fornecido por uma empresa interessada contendo os tópicos e questões abordados durante o curso em segurança no trabalho. Isso resultou em uma nova base, com aproximadamente 950 nós, a qual será chamada neste trabalho por **ExpertNet**.

A escolha por uma abordagem on-line deu-se ao propósito da aplicação ser voltada a EAD. Assim, utilizou-se uma arquitetura baseada em *Web Services* possibilitando o servidor *Web* fazer consultas remotas na *API ConceptNetBr*.

O sítio foi desenvolvido, pelo autor deste trabalho, com o uso da tecnologia *Java Server Pages - JSP*, que utiliza um módulo desenvolvido neste trabalho para se comunicar com o servidor da *ConceptNetBr*. Esse módulo, chamado *OMCS-Br Client*, foi desenvolvido em Java e utiliza o protocolo XML-RPC, mesmo protocolo utilizado pelo servidor da *ConceptNetBr*.

As partes que compõem a arquitetura desenvolvida são as seguintes: o sítio, o servidor Web e o Servidor *ConceptNetBr*. Esse último servidor é responsável pela instanciação da API *ConceptNetBr*, que utiliza duas redes semânticas, a saber, a própria *ConceptNetBr* e a *ExpertNet*, que servirão de domínio base e domínio alvo, respectivamente, durante a construção da analogia. Um diagrama contendo a arquitetura desenvolvida pode ser observado na Figura 18.



**FIGURA 18.** A arquitetura do sistema de apoio para treinamento a distância.

Quando o estudante digita o termo para o qual deseja uma explicação (1), o Servidor *Web* estabelece uma conexão com o Servidor da *ConceptNetBr* através do módulo *OMCS*

*Client* (2). No entanto, como nem sempre o termo pesquisado pelo aluno pode corresponder exatamente ao termo presente na *ExpertNet*, a entrada fornecida pelo aluno passa pelo mesmo processo de normalização (3) realizado para a construção da *ConceptNetBr*, caso o termo pesquisado não seja encontrado na base, duas estratégias são utilizadas a fim de encontrar as variações do termo pesquisado presentes na base.

A primeira estratégia, chamada de expansão, é acionada quando o aluno utiliza expressões curtas na *string* de busca. Ela consiste na pesquisa de nós da rede que tenham a entrada do aluno como *substring*. Por exemplo, se o termo ‘alarme’ não foi encontrado como nó da *ExpertNet*, esta estratégia permite encontrar as variações: “alarme de fogo”, “sirene de alarme”, “alarme nível 1”, entre outras.

Já a segunda estratégia, chamada de quebra, é utilizada caso a primeira estratégia não tenha resultados. Essa estratégia é especialmente útil quando o aluno utiliza grandes expressões como *string* de busca. Ela consiste na quebra da expressão digitada em sintagmas nominais e verbais, realizada pelo analisador sintático Curipira, e a posterior verificação da existência desses sintagmas como nós da *ExpertNet*. Por exemplo, essa estratégia permite que através da *string* de busca “extintor correto para incêndio classe A” sejam encontradas as seguintes variações: “incêndio classe A” e “extintor”.

A aplicação dessas estratégias aumenta a probabilidade do aluno obter resultados para uma pesquisa, pois ele não necessita digitar os termos exatamente como eles constam na *ExpertNet*.

Tendo encontrado os termos a serem utilizados, o Servidor *ConceptNet* realiza a consulta na *ConceptNetBr* e na *ExpertNet* (4) retorna três elementos para o Servidor *Web* (5): todas as relações presentes na *ExpertNet* contendo os variações; uma analogia relacionando o termo a um conceito de senso comum e uma lista dos conceitos que fazem parte do contexto do termo. Em seguida, o Servidor *Web* formata os dados recebidos e os exibe em quatro

seções (6). Um exemplo do formato da resposta apresentada ao aluno pode ser observado na Figura 19.

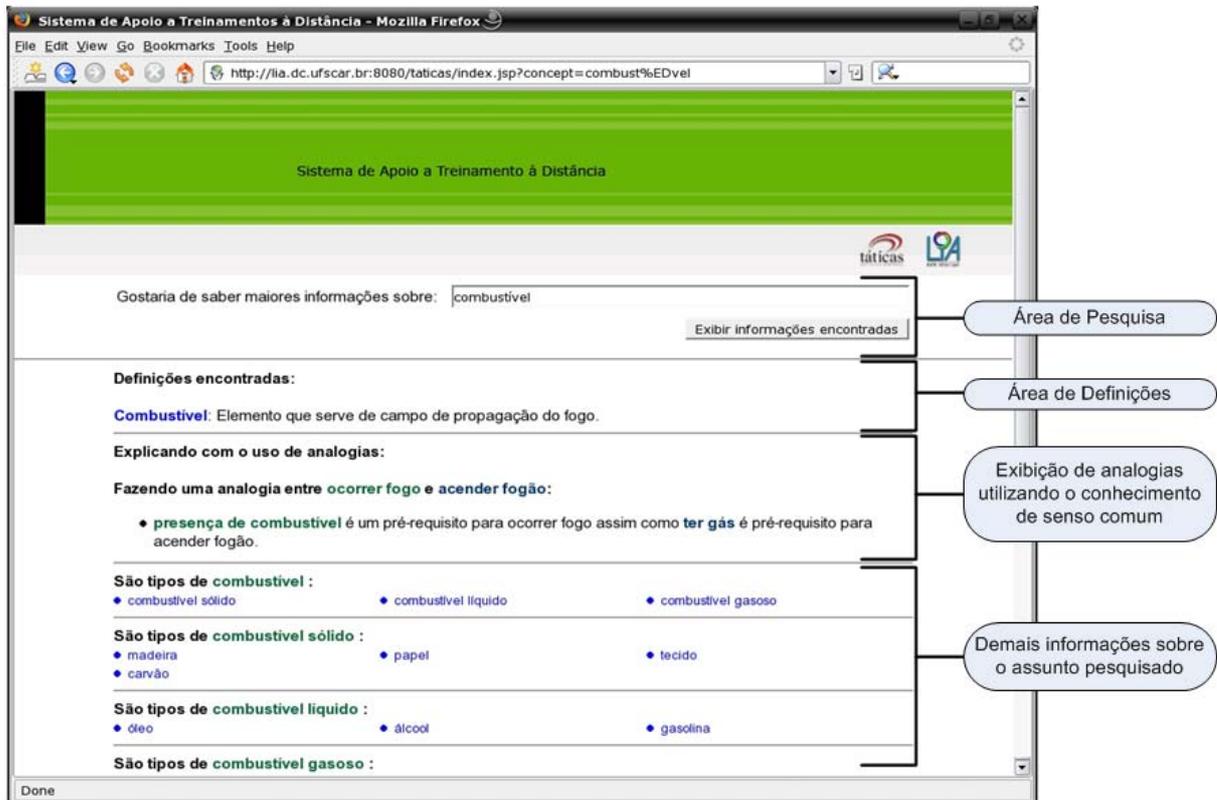


FIGURA 19. Screenshot do sistema de apoio a treinamento à distância.

A primeira seção, rotulada como “Área de Definições”, apresenta as definições encontradas na *ExpertNet* para o termo pesquisado ou sua variante. Essas definições são geradas através das relações do tipo *DefinedAs* presentes na *ExpertNet*.

A segunda seção apresenta uma analogia, que explica o termo pesquisado utilizando o conhecimento de senso comum. Conforme apresentado no capítulo anterior, a analogia gerada é composta por um conjunto de mapeamentos locais. Assim, a analogia é apresentada em língua natural semi-estruturada com uma lista de sentenças na forma: X<relação>Y assim como A<relação>B, onde X e Y pertencem à *ExpertNet* e A e B pertencem à *ConceptNetBr*.

A terceira seção apresenta as relações existentes entre o termo pesquisado com demais conceitos presentes na *ExpertNet*. Os conceitos são apresentados na forma de links: ao clicar em um link uma nova busca é realizada e novos dados são apresentados ao aluno.

A última seção, que pode ser observada na Figura 20, apresenta uma lista de conceitos relacionados com conceito apresentado. Essa lista é obtida pela função de captura de contexto *get\_context* presente na API da *ConceptNetBr*. Ela foi adicionada para encorajar os alunos a explorarem o contexto que eles estão estudando, considerando-se a sua importância do contexto no processo de aprendizagem.

- Conceitos relacionados:**
- combustivel
  - combustivel gasoso
  - eliminacao
  - calor
  - elemento
  - carvão
  - presença
  - serve
  - combustivel sólido
  - líquido
  - abafamento
  - proximidades
  - madeira
  - óleo
  - outros gases
  - campo
  - combustivel líquido
  - gasoso
  - presença de combustivel
  - comburentes
  - papel
  - gasolina
  - triângulo
  - propagação

FIGURA 20. Área de conceitos relacionados

### 5.2.2. Avaliação do Sistema Desenvolvido

No intuito de avaliar o protótipo desenvolvido, foi conduzido um experimento com 24 voluntários. A maioria deles é mestrando, contatados por e-mail, que não possuem familiaridade com o tema de segurança do trabalho. O protótipo foi apresentado aos voluntários, e eles foram instruídos a responder um questionário on-line, que pode ser acessado em [http://lia.dc.ufscar.br:8080/taticas/enquete\\_inicio.jsp](http://lia.dc.ufscar.br:8080/taticas/enquete_inicio.jsp). A tela inicial do sítio desenvolvido para o experimento pode ser observado na Figura 21.

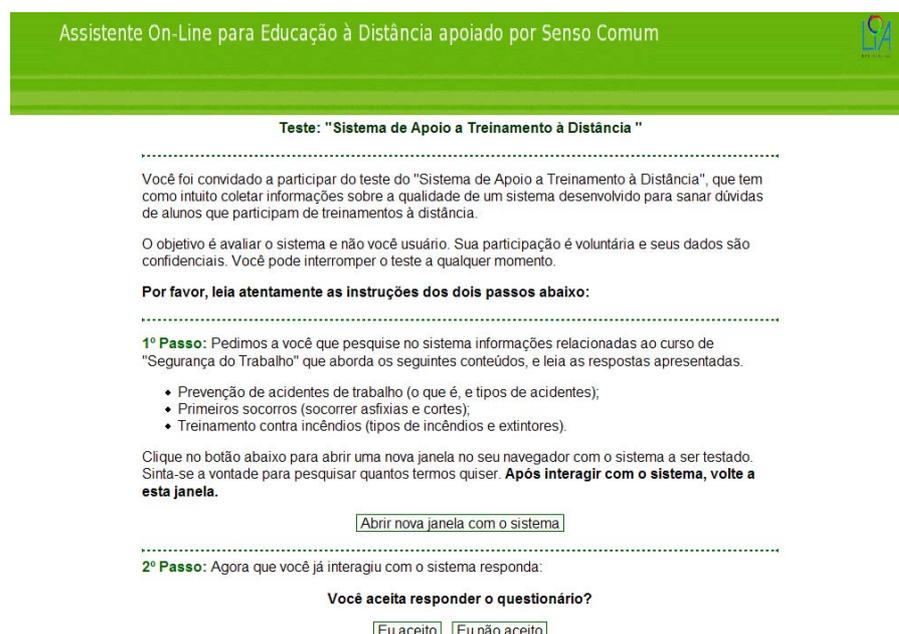


FIGURA 21. Screenshot da tela apresentada aos voluntários antes do experimento.

Antes de começar a responder o questionário, foram apresentadas informações aos voluntários referentes ao objetivo da pesquisa desenvolvida e aspectos éticos do experimento, envolvendo privacidade e confidencialidade, conforme pode ser observado na Figura 21.

O questionário, que pode ser observado na Figura 22, foi dividido em duas partes, a saber: a primeira coleta informações relacionadas com o perfil do voluntário (faixa etária, escolaridade e gênero); a segunda parte apresenta questões referentes à utilização do protótipo.

Assistente On-Line para Educação à Distância apoiado por Senso Comum


---

**Caracterização do usuário:**

1) Qual a sua faixa de idade?

---

2) Qual é seu nível de escolaridade?

---

3) Qual é seu sexo?

---

**Opine sobre o sistema:**

4) Que conceito você dá para a facilidade de utilização do sistema?  
 Muito fácil;  Fácil;  Neutro;  Difícil;  Muito difícil.  
 Se desejar, faça alguma observação:

---

5) Que conceito você dá para a utilidade do sistema em esclarecer dúvidas sobre o tema apresentado?  
 Muito útil;  Útil;  Neutro;  Pouco útil;  Nada útil.  
 Se desejar, faça alguma observação:

---

6) Durante a utilização do sistema, foram apresentadas explicações com o uso de analogias?  
 Sim;  Lembro de algumas;  Lembro vagamente disso;  Não lembro;  Não.  
 Se desejar, faça alguma observação:

---

7) Em caso de terem sido exibidas explicações com o uso de analogia, elas foram úteis para entender o termo pesquisado?  
 Muito úteis;  Úteis;  Pouco úteis;  Nada úteis;  Não foram exibidas analogias.  
 Se desejar, faça alguma observação:

---

8) Se desejar, escreva aqui seus comentários sobre o uso do sistema:

---

FIGURA 22. Questionário utilizado para avaliação do protótipo desenvolvido.

As questões foram avaliadas utilizando uma escala de cinco pontos. Também foi disponibilizado um campo caso os voluntários quisessem expressar algum comentário adicional livremente.

Cerca de 79% dos voluntários consideraram que o sistema foi fácil ou muito fácil de utilizar, conforme mostra a Tabela 6. Alguns comentários fornecidos no campo extra desta questão foram, para exemplificar, os seguintes: “*o sistema é simples de usar e intuitivo*” e “*o sistema é simples de interagir e encontrar coisas*”. É importante ressaltar que um número muito baixo de voluntários considerou o sistema difícil de ser utilizado. Dentre esses voluntários, foram fornecidos comentários, para exemplificar, como os seguintes: “*o sistema poderia exibir uma mensagem ao usuário quando o conceito não foi encontrado*” e “*a maneira que as informações são apresentadas poderia ser mais concisa*”, apontando a necessidade de alguns melhoramentos na interface.

**TABELA 6.** Opinião dos voluntários sobre a facilidade de uso do sistema.

<b>Opinião do usuário</b>	<b>Porcentagem</b>
Muito fácil	25,0 %
Fácil	54,2 %
Neutro	12,5 %
Difícil	8,3 %
Muito difícil	0 %

Quanto à utilidade do sistema em esclarecer dúvidas sobre o tema pesquisado, cerca de 87% dos voluntários consideraram o sistema útil ou muito útil. Uma vez que nenhum usuário considerou o sistema pouco útil ou nada útil, conforme exibido na Tabela 7, pode-se concluir que o sistema atinge o propósito para o qual foi desenvolvido.

**TABELA 7.** Opinião dos voluntários sobre a utilidade do sistema em esclarecer dúvidas.

<b>Opinião do usuário</b>	<b>Porcentagem</b>
Muito útil	20,8 %
Útil	66,7 %
Neutro	12,5 %
Pouco útil	0 %
Nada útil	0 %

Os voluntários também foram questionados sobre se eles se lembravam da presença de analogias nas respostas apresentadas pelo sistema. A grande maioria deles, cerca de 88%, responderam lembrar da presença de analogias. A pergunta seguinte questionou a utilidade da analogia para entender o termo pesquisado.

A Tabela 8 mostra que 67% dos voluntários consideraram a utilização de analogias útil ou muito útil para o entendimento do termo. Considerando os usuários que relataram se lembrar da exibição de analogias, este percentual sobe para 70%. Adicionalmente também foram fornecidos comentários, para exemplificar, dos seguintes tipos: *“eu gostei das analogias porque elas me ajudaram a entender o conceito que está sendo ensinado”* e *“as analogias foram úteis porque elas forneceram claros exemplos sobre os conceitos”*. Esses comentários confirmam o que muitos autores afirmam sobre a utilização de analogias no processo de aprendizagem, demonstrando a relevância da abordagem adotada neste trabalho.

**TABELA 8.** Opinião dos voluntários sobre a utilidade da analogia para entender o termo pesquisado.

<b>Opinião do usuário</b>	<b>Porcentagem</b>
Muito útil	12,5 %
Útil	54,2 %
Pouco útil	29,2 %
Nada útil	0 %
Analogias não foram apresentadas para mim	4,2 %

De uma maneira geral, os dados coletados nesse experimento demonstraram que a maioria dos usuários considera as analogias úteis para o entendimento do conceito apresentado. Alguns comentários adicionais coletados, para exemplificar esse ponto, os seguintes:

- *“o sistema é muito interessante, especialmente pelo fato de utilizar analogias que facilitam o entendimento dos conceitos”;*
- *“uma sugestão é diminuir a redundância presente em algumas definições”;*
- *“As analogias foram satisfatórias e coerentes com o contexto”*

- “*eu tive que gastar algum tempo analisando os dados antes de encontrar respostas para questões específicas*”.

Alguns voluntários mencionaram que algumas vezes foram apresentados dados redundantes, o que aumenta a quantidade de informação exibida, dificultando a localização de informações desejadas pelo usuário. Comentários como esse devem ser considerados para refinar o sistema antes da sua utilização em sistemas reais.

### **5.3. Auxiliando o Professor na Criação de Analogias para o Material Instrucional**

O segundo protótipo está desenvolvido e incorporado ao Cognitor, um editor que auxilia o professor no projeto de material instrucional. O projeto e desenvolvimento de material instrucional contendo objetos de aprendizagem para EAD em ambiente Web pode ser uma tarefa difícil para os professores que têm pouca experiência em interação ou projeto instrucional em ambiente computacional, demandando tempo e recursos muitas vezes não disponíveis e inviabilizando a proposta do desenvolvimento do material. Essa dificuldade acaba gerando uma produção de cursos com materiais instrucionais deficitários, como, por exemplo, documentos de textos com excessiva quantidade de informação, que impedem ou dificultam o processo de aprendizagem dos alunos (Kessler *et al.*, 1999).

Nesse contexto, visando auxiliar o professor na sua tarefa de projetar e editar material instrucional de qualidade para EAD, o LIA iniciou o desenvolvimento da ferramenta Cognitor, que será apresentada nas subseções 5.3.1 a 5.3.4, juntamente com os auxílios desenvolvidos neste trabalho.

#### **5.3.1. Cognitor**

O Cognitor é um editor de material instrucional para aprendizagem eletrônica que oferece auxílio por meio da representação computacional da Linguagem de Padrões para

EAD, a Cog-Learn, como funcionalidades disponibilizadas ao professor. Essas funcionalidades expressam a utilização dos padrões que compõem a LP Cog-Learn, permitindo que o professor projete e elabore o seu material instrucional de maneira automatizada, utilizando as soluções de sucesso para problemas recorrentes encontrados por outros professores durante o projeto de materiais instrucionais.

A LP Cog-Learn (Talarico Neto *et al.*, 2005) foi identificada e formalizada por especialistas em Padrões de Projeto durante a condução de três estudos de caso (Almeida, 2005), que propiciaram a verificação de que a utilização de um conjunto selecionado de estratégias cognitivas (Liebman, 1998), combinadas com padrões de Interação Humano-Computador e pedagógicos, aumenta a usabilidade de materiais instrucionais escritos na forma de hiperdocumentos para EAD, bem como contribui para o aumento na satisfação de uso por parte do usuário final de tais hiperdocumentos – os alunos.

O material instrucional produzido pelo Cognitor pode ser reutilizado em outros contextos de aprendizagem, pois é projetado seguindo os conceitos de objetos de aprendizagem, e pode ser executado em diversas plataformas de EAD, uma vez que pode ser exportado em formato SCORM® (2001) e HTML.

A interface principal do Cognitor é composta por 6 áreas que podem ser observadas na Figura 23: (1) Área de planejamento e organização do material instrucional; (2) Área de planejamento da interação; (3) Área de inserção de mídia e publicação de conteúdo, a saber, som, vídeo, imagens e links externos; (4) Área de edição de página; (5) Área de propriedades da mídia e; (6) Área de controle de objetos.

O Cognitor possui um assistente que permite a criação da estrutura do material instrucional baseado na teoria de mapas de conceitos (Gagné, 1974), gerando automaticamente a estrutura do material para o professor. Estuda-se no LIA mecanismos baseados no conhecimento de senso comum, para auxiliar o professor na tarefa de construção

do mapa de conceitos (Anacleto *et al.*, 2007).

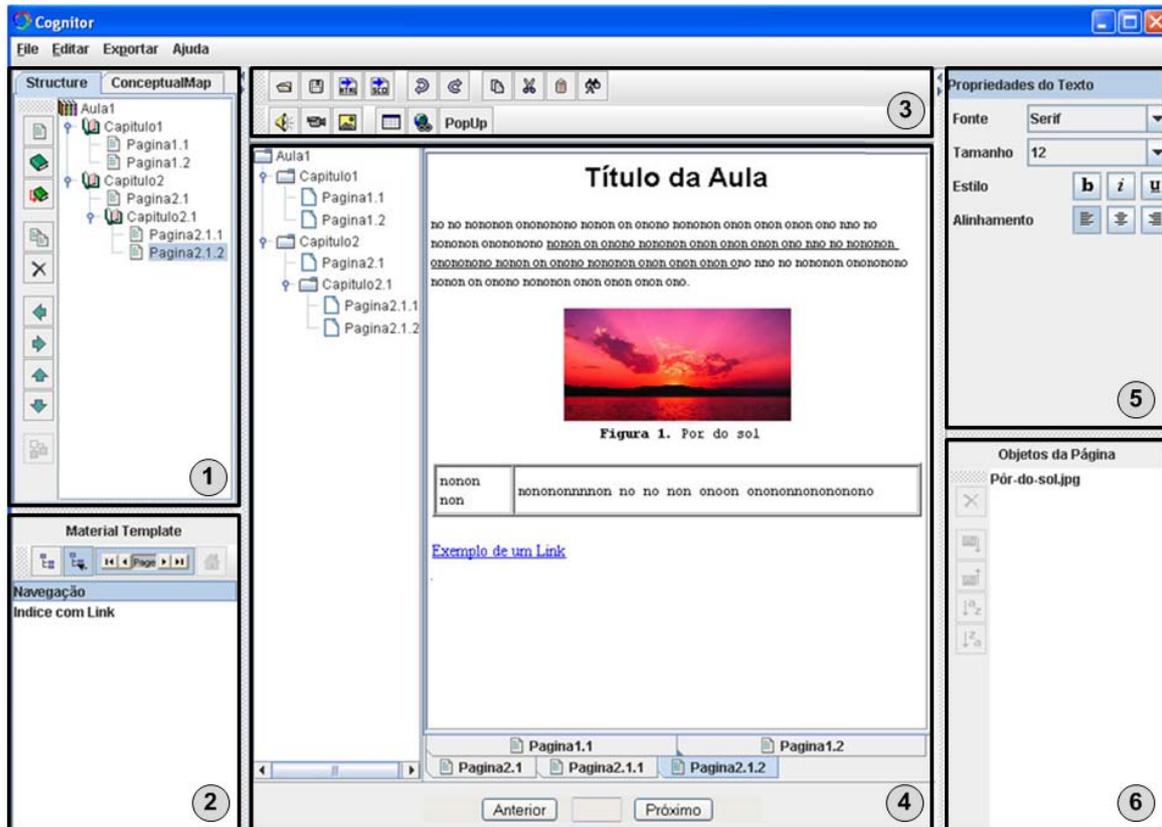


FIGURA 23. Interface principal da ferramenta Cognitor.

### 5.3.2. O padrão Correlação

O software que foi implementado na forma de um assistente para a geração de analogias é a representação computacional de um padrão presente na LP Cog-Learn, chamado Correlação. Esse padrão será apresentado a seguir no seguinte formato: nome do padrão, contexto em que pode ser aplicado, problema recorrente nesse contexto, solução de sucesso para esse problema e outros padrões com o qual está relacionado.

**Nome do padrão** - Correlação.

**Contexto** - Ao ensinar um tópico complexo fora da experiência normal do aluno, encontre uma analogia complexa e consistente para o tópico que está sendo ensinado. O contexto base da analogia deve ser de conhecimento dos alunos.

**Problema** - Como fazer com que os alunos vejam rapidamente como o tópico se relaciona com os objetivos maiores e entendam como os conceitos se relacionam?

**Solução** - Utilize a Estratégia Cognitiva Metáforas e Analogias. Crie uma Metáfora que seja consistente com o tópico que está sendo ensinado. Forneça aos alunos uma maneira rápida de pensar sobre o tópico.

**Padrões Relacionados** – Sedimentação.

### 5.3.3. Implementação do Assistente Para Geração de Analogias

Através da apresentação do padrão Correlação, pode-se identificar claramente a potencialidade da implementação de um mecanismo que utilize o algoritmo utilizado nesse trabalho como sendo a representação computacional desse padrão dentro do editor de material instrucional Cognitor.

Porém, para a execução do algoritmo é necessária uma base de conhecimento contendo o conhecimento especialista tema do material instrucional desenvolvido. Poder-se-ia pensar na construção prévia da base por um engenheiro do conhecimento, assim como foi feito no protótipo apresentado anteriormente. Porém, a construção prévia dessa base é inviável, pois o Cognitor é um editor que pode ser utilizado para o projeto de materiais instrucionais nos mais diversos domínios, diferentemente do protótipo anterior que, apesar de poder ser utilizado em diferentes domínios, foi instanciado para um domínio específico.

Outra possibilidade é o Cognitor fornecer meios para que o professor forneça as informações mínimas necessárias para a construção da analogia. Provavelmente a maneira mais fácil para o professor fazer isso seria o Cognitor processar o texto em linguagem natural que está sendo editado e construir automaticamente uma *MiniExpertNet*. Porém, a tarefa de entendimento de linguagem natural não é simples e tem sido um dos grandes temas pesquisados pela IA desde seu surgimento, fugindo assim do escopo deste trabalho, que está focado na construção de analogias para a educação utilizando o conhecimento de senso comum.

Assim, foi adotada nesse protótipo a mesma abordagem do projeto OMCS para a

coleta de dados. Para isso, foi implementada uma interface que permite ao professor fornecer os dados de uma maneira semi-estruturada com o uso de gabaritos. A arquitetura do segundo protótipo desenvolvido pode ser observada na Figura 24.

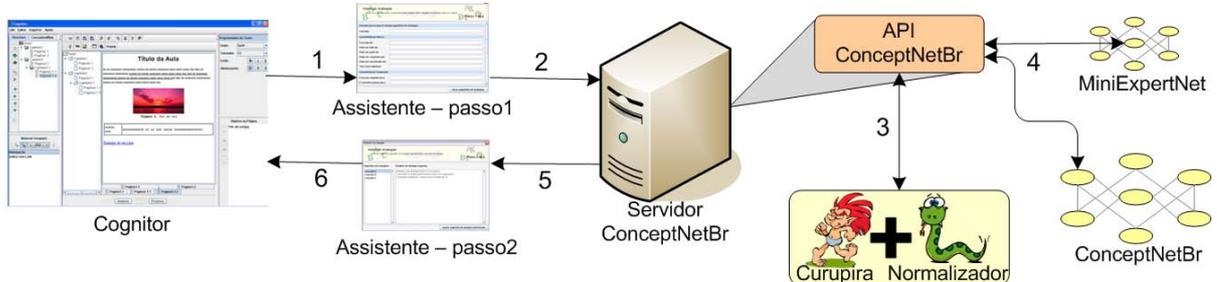


FIGURA 24 - Arquitetura do protótipo do assistente que sugere analogias ao professor.

A interface para coleta dos dados da *MiniExpertNet* é apresentada como primeiro passo de um assistente, que pode ser observado na Figura 25. Esse assistente foi adicionado ao Cognitor como sendo a representação computacional do padrão Correlação. No primeiro passo, (1) o professor fornece as informações sobre o termo para o qual deseja sugestões de analogias. São fornecidos 17 campos ao professor, no qual ele pode expressar as 17 possíveis relações presentes na *ConceptNet*, desconsiderando as relações do tipo K-line.

A janela 'Inserir Analogia' apresenta o seguinte conteúdo:

- Título:** Inserir Analogia
- Instruções:** Descreva o conceito para o qual deseja uma analogia. Você não é obrigado a preencher todos os campos. Para inserir mais de uma resposta por campo basta separá-las por vírgula.
- Progresso:** Passo 1 de 2
- Seção 1:** Conceito para o qual se deseja sugestões de analogias:
  - Conceito:
- Seção 2:** Características Funcionais:
  - É usado para:
- Seção 3:** Características Comportamentais:
  - É capaz de:
- Seção 4:** Características Temporais:
  - é um pré-requisito para:
  - É o primeiro passo para:
  - É um passo para:
  - É o último passo para:
- Seção 5:** Características Físicas:
- Botão:** Gerar possíveis analogias

FIGURA 25. Interface do primeiro passo do assistente para criação de analogias.

O professor não é obrigado a preencher todos os campos. Ele deve preencher somente os campos que considerar mais convenientes e importantes. Também é possível ao professor adicionar mais de uma resposta por campo, bastando para isso que ele separe as respostas por vírgula. Uma vez preenchidos os campos desejados, o professor clica no botão “Gerar sugestões de analogias”. Em seguida, os dados são convertidos para o formato da base OMCS-Br para que seja possível a construção da MiniExpertNet

Após a conversão, os dados são enviados para o Servidor da *ConceptNet* (2), que os normaliza os dados enviados (3) e, através do mesmo módulo utilizado no protótipo anterior, sendo gerada assim uma *MiniExpertNet*, que é então utilizada para a geração das analogias (4).

Neste protótipo o algoritmo retorna como resposta mais de uma analogia. Portanto, serão retornados os mapeamentos globais que obtiverem as melhores avaliações. O método utilizado para exibir as analogias retornadas é similar ao utilizado no protótipo anterior, conforme se pode observar na Figura 26, que apresenta a interface do passo dois do assistente, exibindo uma resposta fictícia com analogias para um termo X.

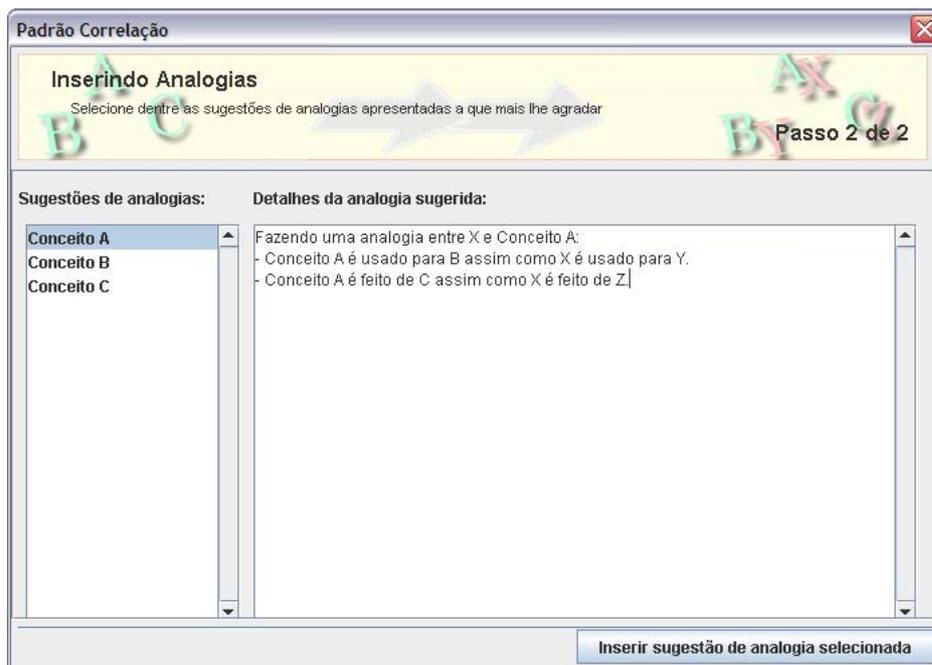


FIGURA 26. Interface do segundo passo do assistente para criação de analogias.

O professor então pode visualizar as analogias geradas e decidir qual mais o agrada (5), adicionando-a automaticamente no material instrucional (6). Caso o professor não considere nenhuma das analogias apresentadas útil, basta fechar a janela do assistente e retornar à edição de seu material instrucional.

Se o assistente não encontrar nenhuma analogia para o termo solicitado, será exibida uma mensagem orientando a retornar ao primeiro passo e descrever mais detalhadamente o conceito para o qual deseja uma analogia.

#### 5.3.4. Análise dos Estudos de Casos Reais Simulados no Segundo Protótipo

Com a finalidade de analisar a validade do protótipo desenvolvido, serão apresentados estudos de casos reais simulados, que demonstram o funcionamento do assistente em um caso próximo à sua utilização no mundo real.

##### Primeiro Estudo de Caso Real Simulado

No primeiro caso, será simulado o professor no contexto do desenvolvimento de um material instrucional sobre o tema “educação sexual”. Com a finalidade de melhorar a explicação, solicita uma analogia para o termo “camisinha”, preenchendo os campos do assistente conforme ilustrado na Figura 27, a seguir:

FIGURA 27. Professor inserindo dados sobre camisinha no primeiro passo do assistente.

Os dados inseridos no passo 1 do assistente são convertidos, nos padrões do projeto OMCS-Br para seguinte o formato:

```
(UsedFor "camisinha" "proteger" "f=1;i=1")
(PrerequisiteEventOf "camisinha" "ter relações sexuais" "f=1;i=1")
```

Ao clicar no botão “Gerar Possíveis Analogias”, o assistente vai para o passo 2, ilustrado na Figura 28, exibindo ao professor cinco sugestões de analogias, a saber: “policia militar”, “faixa de segurança”, “portaria”, “capa de chuva” e “balcão”.

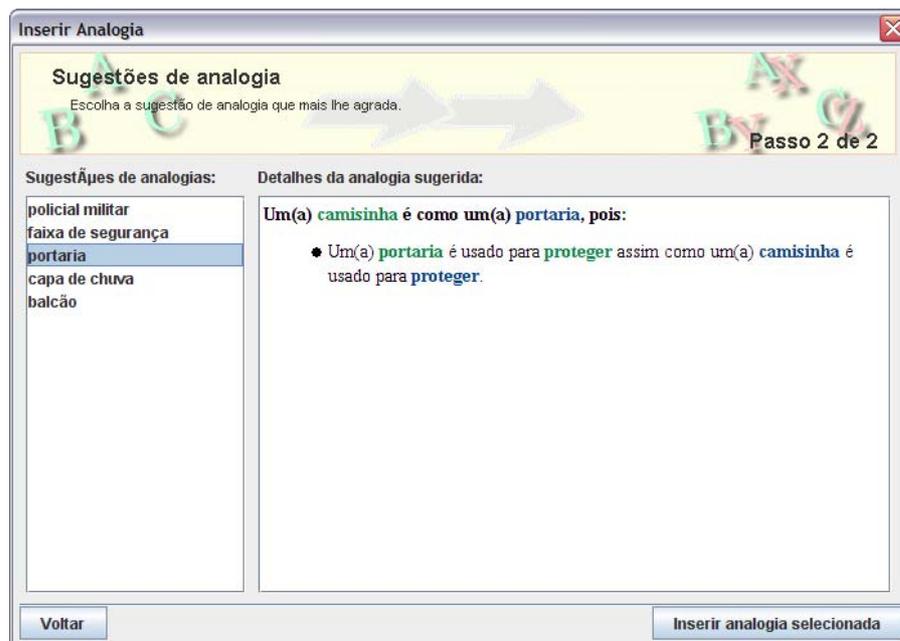


FIGURA 28. Segundo passo do assistente exibindo analogias para o termo ”camisinha”.

Percebe-se que todos os termos retornados como sugestões de analogias são utilizados para proteger algo. No ponto de vista de um julgador humano, nesse caso o autor deste trabalho, a analogia mais útil seria “portaria”, pois, apesar de não ser apresentado explicitamente no texto da analogia, também remete à idéia de barrar a entrada, assim como a camisinha faz.

## Segundo Estudo de Caso Real Simulado

No primeiro caso, será simulado o professor no contexto do desenvolvimento de um material instrucional sobre o tema “saúde”. Com a finalidade de melhorar a explicação, solicita uma analogia para o termo “aparelho de pressão”, preenchendo os campos do assistente conforme ilustrado na Figura 29, a seguir:

FIGURA 29. Professor inserindo dados sobre aparelho de pressão no primeiro passo do assistente.

Apesar de não serem exibidos na figura, também foi preenchido o campo “É um tipo de” com “instrumento”. Os dados inseridos no passo 1 do assistente são convertidos, nos padrões do projeto OMCS-Br para seguinte o formato:

```
(IsA "aparelho de pressão" "instrumento" "f=1;i=1")
(UsedFor "aparelho de pressão" "medir" "f=1;i=1")
(UsedFor "aparelho de pressão" "inferir" "f=1;i=1")
(UsedFor "aparelho de pressão" "verificar" "f=1;i=1")
```

Ao clicar no botão “Gerar Possíveis Analogias”, o assistente vai para o passo 2, ilustrado na Figura 30, exibindo ao professor seis sugestões de analogias, a saber: “multímetro”, “esquadro de plástico”, “medidor”, “pipeta”, “multímetro” e “grama”.

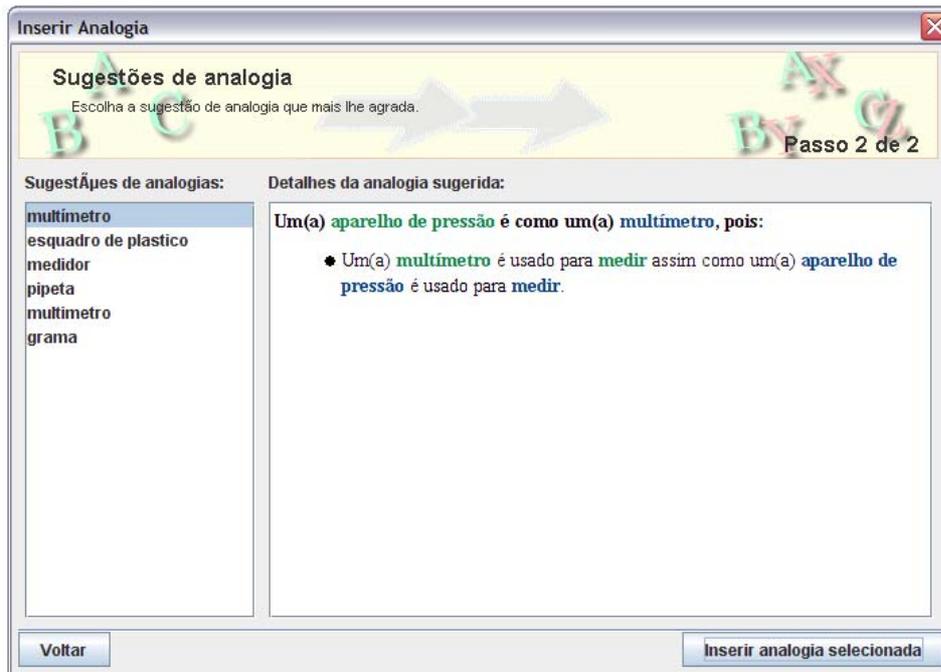


FIGURA 30. Segundo passo do assistente exibindo analogias para o termo "aparelho de pressão".

Pode-se perceber nesse caso que a maioria das analogias foram baseadas em coisas que servem para medir. No ponto de vista de um julgador humano, nesse caso o autor deste trabalho, a analogia mais útil seria "multímetro", que também é um instrumento que é utilizado para realizar medições.

Fazem-se necessárias algumas observações neste estudo de caso simulado. Primeiramente, os termos sugeridos como sendo de conhecimento de senso comum para as analogias, apresentaram aparelhos técnicos, cujos nomes e características podem não ser de senso comum, como por exemplo, "multímetro", "esquadro de plástico" e "pipeta". Essas analogias podem, ou não, serem úteis, dependendo do perfil dos alunos que irá utilizar o material.

Pode-se também observar a presença de um erro de ortografia na base utilizada, e a conseqüente não unificação dos termos "multímetro" e "multimetro". Indicando falhas no processo de revisão humano realizado pelo projeto OMCS-Br.

A última sugestão de analogia, a saber, "grama" se diferencia das outras por ser uma comparação do tipo analogia, enquanto todas as outras são do tipo similaridade literal.

Conforme pode ser observado na Figura 31. No entanto, apesar de terem sido apresentadas mais informações, quando comparadas às similaridades literais apresentadas, esta sugestão de analogia pode não ser considerada útil para o material instrucional, do ponto de vista de um revisor humano.

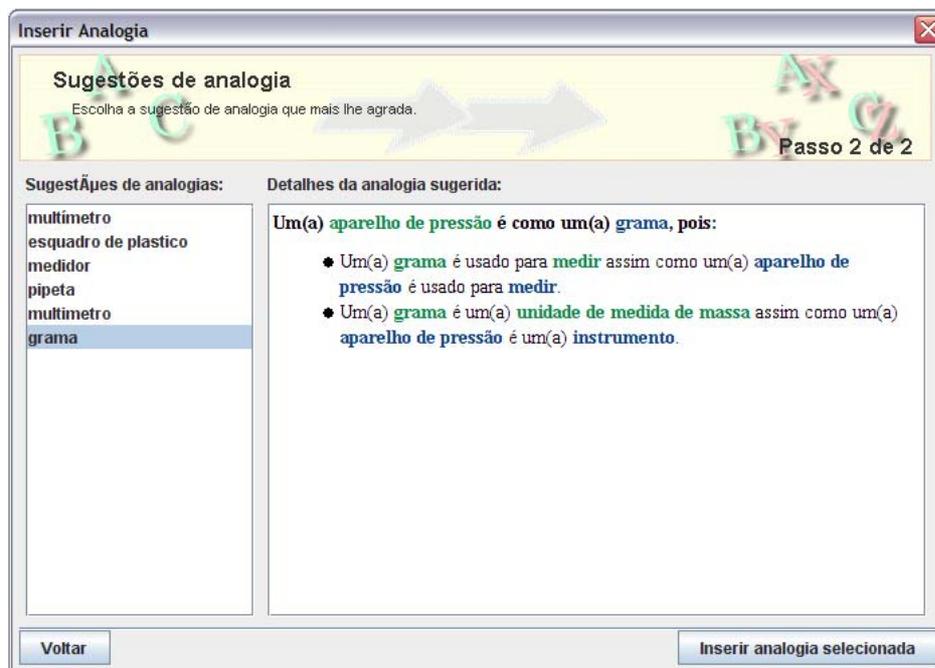


FIGURA 31. Segundo passo do assistente exibindo analogias para o termo "aparelho de pressão".

#### 5.4. Considerações finais

Este capítulo apresentou dois protótipos desenvolvidos que utilizam o algoritmo apresentado no capítulo anterior. Esses protótipos foram construídos com o intuito de validar a abordagem proposta neste trabalho para a criação de analogias entre o conhecimento de senso comum e o conhecimento especializado e a validade de sua utilização em aplicações educacionais.

Os resultados dos experimentos com o primeiro protótipo se mostraram satisfatórios em mostrar a validade da abordagem desenvolvida neste trabalho. No entanto, vale ressaltar que as aplicações desenvolvidas são protótipos que visam apenas à validação deste trabalho. Portanto, necessitariam passar por refinamentos antes de serem utilizados como produtos em aplicações do mundo real.

Os estudos de casos reais simulados aplicados ao segundo protótipo mostram a potencialidade da aplicação desenvolvida, apesar de nem sempre exibir boas sugestões de analogias do ponto de vista de um revisor humano. Além da implementação de futuras melhorias no processo de avaliação das analogias geradas, espera-se que o número de analogias úteis aumente juntamente com o crescimento da base, aumentando os pesos dos conhecimentos de senso comum e diminuindo o peso dos conhecimentos especialistas presentes na base.

Uma possibilidade de expansão desse segundo protótipo é o professor fornecer o perfil do aluno para o qual o material está sendo desenvolvido. Assim, seria possível a construção de analogias considerando o conhecimento de senso comum dos colaboradores de um determinado perfil. Aumentando, assim, as chances da geração de analogias mais contextualizadas ao perfil do aluno.

O próximo capítulo apresenta as conclusões deste trabalho, as dificuldades a serem superadas para a utilização dos mecanismos desenvolvidos neste trabalho por produtos em aplicações reais e sugestões de trabalhos futuros.

## **CAPÍTULO 6 - CONCLUSÕES**

---

### **6.1. Conclusões**

Através da pesquisa desenvolvida neste trabalho pôde-se concluir que é possível e válida a abordagem da utilização de conhecimento de senso comum, armazenado em uma base que está sendo construída colaborativamente por voluntários da Web, para a geração de analogias cujo objetivo foi explicar conhecimento especialista, no contexto de educação a distância. Assim, o aprendiz pode, através do uso de analogias relacionadas com a sua cultura e contexto, ou seja, ao seu conhecimento de senso comum, compreender mais facilmente conceitos de conhecimento especializado que sejam análogos de alguma forma com coisas que já são de seu conhecimento ou fazem parte de seu dia-a-dia.

De maneira geral, os dados coletados no experimento desenvolvido no contexto de segurança no trabalho mostram que a maioria dos usuários considerou que as analogias geradas foram úteis no processo de aprendizado. Considera-se também que os resultados dos estudos de casos reais simulados foram suficientes para mostrar a potencialidade da abordagem adotada.

### **6.2. Análise Crítica**

Um ponto importante de ser observado é o fato do experimento realizado com o primeiro protótipo desenvolvido foi realizado com a participação de estudantes e não com empregados de uma empresa, que são os usuários finais para os quais se destina o protótipo. Experimentos com empregados reais não foram possíveis devido a ausência da empresa conveniada com o LIA. Porém, considera-se que o experimento realizado com estudantes mostrou bons resultados, indicando a validade da abordagem apresentada neste trabalho.

O tamanho da base de senso comum também é outro ponto importante de se ressaltar na análise crítica, uma vez que a base utilizada é composta por 150 mil fatos de senso comum. Esse número está longe dos milhões de fatos necessários para simular o conhecimento de

senso comum de uma pessoa segundo apontado por Minsky (1983). A base atual assim, pode não conter a melhor os dados suficientes para a geração de uma analogia, que possa ser considerada boa para um revisor humano. Porém, acredita-se que com o crescimento do projeto e, conseqüentemente, do número de fatos na base, esses efeitos relativos ao tamanho da base sejam minimizados.

### **6.3. Dificuldades a serem superadas**

A restrição quanto ao trabalho computacional e uso de memória exigidos para a geração de analogias aumentar de uma maneira não linear é um obstáculo a ser vencido, considerando os benefícios da utilização de uma base maior e o rápido tempo de resposta exigido em aplicações Web. Assim observa-se a necessidade de realizar novas melhorias no desempenho computacional do algoritmo de geração de analogias para tornar sua utilização viável em aplicações reais.

Considerando os resultados obtidos pelo questionário aplicado aos participantes do experimento, percebe-se a necessidade da eliminação de fatos redundantes que estão sendo apresentados no protótipo do assistente desenvolvido, pois aumentam a quantidade de informações na tela, potencialmente atrapalhando a visibilidade de certos resultados que o usuário deseja encontrar. Os comentários obtidos nos questionários deverão ser analisados no refinamento dos requisitos do sistema, para futura evolução do protótipo aqui apresentado.

### **6.4. Publicações obtidas**

A seguir, a lista das publicações relacionadas a este trabalho das quais este pesquisador participou como autor:

#### **Artigo em periódico**

ANACLETO, J. C. ; CARVALHO, A. F. P. DE .; GODOI, M. S. ; NERIS, V. P. A. ; TALARICO NETO, A. ; Cognitor: um Framework baseado na Linguagem de Padrões Cog-Learn. In: **Revista Brasileira de Informática na Educação (RBIE)**, 2007.

### **Artigo Completo em Anais de Congressos**

ANACLETO, J. C. A; GODOI, M. S.; CARVALHO, A. F. P. DE; LIEBERMAN, H. A  
Common Sense-Based On-line Assistant for Training Employees (Prelo). In: Conference-  
Socially Responsible Interaction (INTERACT 2007), 2007.

CARVALHO, A. F. P. DE ; ANACLETO, J. C. ; LIEBERMAN, H. ; GODOI, M. S. ; ZEM-  
MASCARENHAS, S. . Using Common Sense for Planning Learning Activities. In:  
Workshop on Common Sense and Intelligent User Interfaces (CSIUI 2007), 2007, Honolulu.  
**Proc. CSIUI 2007**, 2007.

ANACLETO, J. C. ; LIEBERMAN, H. ; TSUTSUMI, M. ; NERIS, V. P. A. ;CARVALHO,  
A. F. P. DE ; ESPINOSA, J. ; ZEM-MASCARENHAS, S. ; GODOI, M.S. . Can common  
sense uncover cultural differences in computer applications?. In: IFIP World Computer  
Conference (WCC2006), 2006, Santiago. **Artificial Intelligence in Theory and Practice**.  
New York : Springer-Verlag, 2006. v. 217. p. 1-10.

ANACLETO, J. C. ; LIEBERMAN, H. ; CARVALHO, A. F. P. DE ; NERIS, V. P. A. ;  
GODOI, M. S. ; TSUTSUMI, M. ; ESPINOSA, J. ; TALARICO NETO, A. ;  
ZEMMASCARENHAS,S. . Using common sense to recognize cultural differences. In: Ibero-  
American Artificial Intelligence Conference/Brazilian Artificial Intelligence Symposium  
(IBERAMIA/SBIA 2006), 2006, Ribeirão Preto. Advances in Artificial Intelligence -  
IBERAMIA-SBIA 2006. **Lecture Notes in Artificial Intelligence**. Heidelberg : Springer-  
Verlag, 2006. v. 4140. p. 370-379.

ANACLETO, J. C. ; CARVALHO, A. F. P. DE ; NERIS, V. P. A. ; GODOI, M. S. ; ZEM-  
MASCARENHAS, S. ; TALARICO NETO, A. . How Can Common Sense Support  
Instructors with Distance Education?. In: XVII Simpósio Brasileiro de Informática na

Educação (SBIE 2006), 2006, Brasília. **Anais...** 2006. v. 1. p. 217-226.

TALARICO NETO, A. ; ANACLETO, J. C. ; NERIS, V. P. A. ; GODOI, M. S. ; CARVALHO, A. F. P. DE . Cognitor: um Framework baseado na Linguagem de Padrões Cog-Learn. In: XVII Simpósio Brasileiro de Informática na Educação (SBIE 2006), 2006, Brasília. **Anais...** 2006. v. 1. p. 529-538.

TALARICO NETO, A. ; ANACLETO, J. C. ; NERIS, V. P. A. ; GODOI, M. S. ; CARVALHO, A. F. P. DE . Framework baseado na Linguagem de Padrões Cog-Learn para apoio a criação de objetos de aprendizagem (no prelo). In: XII Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia 2006), 2006, Natal. **Anais...** Porto Alegre : Sociedade Brasileira da Computação, 2006. v. 1. p. 128-137.

### **Resumo Expandido em Anais de Congressos**

ANACLETO, J. C. ; CARVALHO, A. F. P. DE ; NERIS, Vania Paula de Almeida ; GODOI, M. DE S. ; ZEMMASCARENHAS, Silvia Helena ; TALARICO NETO, Americo ; LIEBERMAN, Henry . Applying Common Sense to Distance Learning: the Case of Home Care Education. In: VII Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais (IHC 2006), 2006, Natal. **Anais Estendidos do VII Simpósio Sobre Fatores Humanos em Sistemas Computacionais**. Porto Alegre : Sociedade Brasileira de Computação, 2006. v. 1. p. 1-4.

ANACLETO, J. C. ; CARVALHO, A. F. P. DE ; GODOI, M. S. ; NERIS, V. P. A. ; ZEMMASCARENHAS, S. ; TALARICO NETO, A. . Common Sense Knowledge Supporting Education. In: III Workshop de Tecnologia da Informação no Desenvolvimento da Internet Avançada ( WSTIDIA 2006), 2006, São Paulo. **Proc. III Workshop TIDIA**. São Carlos : ICMS/USP, 2006. v. 1. p. 127-129.

ANACLETO, J. C. ; GODOI, M. S. ; CARVALHO, A. F. P. DE ; TALARICO NETO, A. ; NERIS, V. P. A. . Supporting Distance Learning Using an On-Line Help Assistant with Common Sense Knowledge. In: III Workshop TIDIA, 2006, São Paulo. III Workshop de Tecnologia da Informação no Desenvolvimento da Internet Avançada ( WSTIDIA 2006), 2006, São Paulo. **Proc. III Workshop TIDIA**. São Carlos : ICMS/USP, 2006. v. 1. p. 130-132.

TALARICO NETO, A.; ANACLETO, J. C. ; GODOI, M. DE S. ; CARVALHO, A. F. P. DE ; NERIS, V. P. de A.; TAKAHAMA, M. ; PEREIRA, A. J. . Framework based on Cog-Learn Pattern Language to Support the Design of Learning Objects. In: III Workshop de Tecnologia da Informação no Desenvolvimento da Internet Avançada (WSTIDIA 2006), 2006, São Paulo. **Proc. III Workshop TIDIA**. São Carlos: ICMC/USP, 2006. v. 1. p. 1-3.

## 6.5. Trabalhos futuros

Como trabalhos futuros, em relação ao algoritmo desenvolvido, faz-se necessária uma formalização que justifique os valores adotados utilizados na avaliação as analogias , que neste trabalho foram obtidos empiricamente.

Considerando o primeiro protótipo, uma melhor abordagem na exibição dos resultados do primeiro protótipo para eliminar as redundâncias presentes nos resultados pode ser visto como um trabalho futuro. Além da realização de testes com voluntários pertencentes ao perfil do usuário final para o qual o protótipo foi desenvolvido, dos quais podem resultar mais dados para análise, permitindo, mais precisamente, encontrar problemas e novas potencialidades a serem exploradas.

Considerando o segundo protótipo, a realização de testes com professores reais poderia facilitar a identificação de pontos a serem melhorados. Pode-se pensar também, como trabalhos futuros, na utilização dos dados inseridos pelos professores no primeiro passo do

assistente para a construção de uma base de conhecimento especialista, facilitando futuras consultas. Através do armazenamento de quais analogias foram inseridas pelo professor no material instrucional, vê-se a possibilidade de gerar um sistema de reforço que melhore o mecanismo de classificação das analogias a serem apresentadas.

Considerando os trabalhos realizados no LIA, podemos vislumbrar a integração mais mecanismos que façam uso do conhecimento de senso comum, através da da ConceptNetBr, no editor de material instrucional Cognitor, cujo foco principal é o apoio ao professor durante a edição de material de aprendizagem a ser disponibilizado na Web. Dentre esses mecanismos está o segundo protótipo, que foi apresentado na Seção 5.3.

Acredita-se que o protótipo em desenvolvimento poderá sugerir analogias ao professor que então geraria o conteúdo usando o Cognitor, de tal forma a encorajar o uso de um vocabulário adequado ao aprendiz, usando analogias, e apresentar o conhecimento de forma que o aprendiz possa entendê-lo mais facilmente.

De acordo com Almeida (2005), essas características aumentam a usabilidade do hiperdocumento, fato que também será estudado mais detalhadamente em trabalhos futuros.

## BIBLIOGRAFIA

---

- ALMEIDA, V. P. **Estratégias cognitivas para o aumento da qualidade do hiperdocumento para educação à distância**. 2005. 158 p. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de São Carlos, São Carlos.
- ANACLETO, J. C. ; LIEBERMAN, H. ; TSUTSUMI, M. ; NERIS, V. P. A. ; CARVALHO, A. F. P. DE ; ESPINOSA, J. ; ZEM-MASCARENHAS, S. ; GODOI, M.S. .Can common sense uncover cultural differences in computer applications?. In: IFIP World Computer Conference (WCC2006), 2006, Santiago. **Artificial Intelligence in Theory and Practice**. New York : Springer-Verlag, 2006. v. 217. p. 1-10.
- ANACLETO, J. C. ; CARVALHO, A. F. P. DE .; GODOI, M. S. ; NERIS, V. P. A. ; TALARICO NETO, A. ; Cognitor: um Framework baseado na Linguagem de Padrões Cog-Learn. In: **Revista Brasileira de Informática na Educação (RBIE)**, 2007.
- ASHWORTH, E. Jennifer, "Medieval Theories of Analogy", **The Stanford Encyclopedia of Philosophy (Winter 2004 Edition)**, Edward N. Zalta (ed.), Disponível em: <<http://plato.stanford.edu/archives/win2004/entries/analogy-medieval/>>.
- AUSUBEL, D.P. (1976) Significado y aprendizaje significativo. In: **Psicología educativa: un punto de vista cognoscitivo**. Mexico: Editorial Trillas, 1976. p. 55-107.
- BECKMAN, P. (2002). **Strategy instruction: ERIC clearinghouse on disabilities and gifted education arlington**. Disponível em: <<http://eric.ed.gov/ERICWebPortal/recordDetail?accno=ED474302>> Acesso em: mar. 2007
- DOUBE, W. Distance Teaching Workloads. In: Technical Symposium on Computer Science Education. **ACM SIGCSE Proceedings**. P. 347-351. Austin,2000.
- FALKENHAINER, B., et. at. The structure-mapping engine: algorithm and examples, **Artificial Intelligence** 41 (1990) 1-63. Disponível em: <http://citeseer.ist.psu.edu/falkenhainer89structuremapping.html>. Acesso em: jun. 2007.
- FERREIRA, A. M. ; PEREIRA, E. N. ; ANACLETO, J. C. ; FABRO, J. A. ; PENTEADO, R. A. D. . Contextualização de jogos educacionais utilizando conhecimento de senso comum. In: SBIE, 2007, São Paulo. SBIE2007, 2007. v. 1. p. 1-1.
- FREINET, C. **Education through work: A model for child centered learning**. Edwin Mellen Press, New York 1993. 438 p.
- FREIRE, P. **Pedagogia da autonomia: saberes necessários à prática educativa**. 31 ed. São Paulo: Paz e Terra, 1996.

GAGNÉ, R. M. (1974). **The Conditions of Learning**. 3. ed. Holt, Rinehart e Winston, 1974.

GENTNER, D. **Structure-mapping: A theoretical framework for analogy**. In: *Cognitive Science* 7(2), (1983): 155-170.

HOUAISS, A. **Dicionário Houaiss da língua portuguesa**. Editora Objetiva, 2001. 3008 p.

KESSLER, G. *et al.*. The Web Can Be Suitable for Learning. **Computer**, 32(2), pp. 114-115.

LENAT, D. B. CYC: a large-scale investment in knowledge infrastructure. **Communications of the ACM**, US (New York, NY): ACM Press, v. 38, n. 11, nov. 1995.

LIBERMAN, H; LIU, H. Adaptive linking between text and photos using common sense reasoning. In: Adaptive Hypermedia And Adaptive Webbased Systems (Ah), International Conference, 2., 2002, Spain (Malaga). **Proceedings...** Berlin: Springer, Lecture Notes in Computer Science, v. 2347, 2002. pp. 2-11.

LIEBERMAN H.; SCHMANDT C. **Common sense reasoning for multi-lingual communication**. Massachusetts: MIT/Media Laboratory Software Agents Group, 2005.

LIEBMAN, J. Teaching Operations Research: Lessons from Cognitive Psychology.

**Interfaces**, v. 28, n. 2, 1998, pp.104-110.

LIU, H. MontyLingua: An end-to-end natural language processor with common sense. 2004. Disponível em: <[web.media.mit.edu/~hugo/montylingua](http://web.media.mit.edu/~hugo/montylingua)>. Acesso em: mai. 2007.

LIU, H.; SINGH P. ConceptNet: a practical commonsense reasoning toolkit. **BT Technology Journal**, v. 22, n. 4, p. 211-226, 2004.

LIU, H.; SINGH, P. Commonsense reasoning in and over natural language. In: International Conference On Knowledge-Based Intelligent Information & Engineering Systems (KES'04), 8., 2004, New Zealand (Wellington). **Proceedings...** Berlin: Springer-Verlag, Lecture Notes in Computer Science, vol. 3214, 2004.

MARTINS, R. T. *et al.*. Curupira: um parser funcional para o português. **Relatório Técnico**. São Carlos: NILC, 2002. NILC-TR-02-26

McCARTHY, J. Programs with common sense. In: Symposium On The Mechanization Of Thought Processes, 1959. **Proceedings...** London: Her Majesty's Stationery Oce. v.1,p.77-84, 1959. Disponível em: <<http://www.formal.stanford.edu/jmc/mcc59.pdf>>. Acesso em: jun. 2007.

MINSKY, M. **K-lines: the theory of memory**. Massachusetts Institute of Technology. Report AIM-516, 1979. Disponível em: <<https://dspace.mit.edu>>. Acesso em: jun. 2006.

MINSKY, M. **The Society of Mind**. New York: Simon and Schuster, 1988.

MATUSZEK, C.; WITBROCK, M.; KAHLERT, R.; CABRAL, J.; SCHNEIDER D. ; SHAH, P.; LENAT, D. B (2006). Searching for Common Sense: Populating Cyc from the Web. In: NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, **Proceedings...** p 20., 2006, Pittsburgh (USA).

MUELLER, E. T. **Natural language processing with ThoughtTreasure**. New York: Signiform, 1998.

MUELLER, E. T. **ThoughtTreasure: A natural language/commonsense platform**. 2007. Disponível em: < <http://www.signiform.com/tt/html/overview.htm>>. Acesso em: jun. 2007.

MUNIZ, M. C. M. **A construção de recursos lingüístico-computacionais para o português do Brasil**: o projeto de Unitex-PB. Dissertação de Mestrado. Instituto de Ciências Matemáticas de São Carlos, USP. 72p. 2004.

NERIS, V. P. de A.; ANACLETO, J. C. ; TALARICO NETO, Americo . Cognitive Strategies to the Increase of Hyper Document Quality to Distance Learning. In: **CLIHC - Conferência Latino-Americana de Interação Humano-Computador**, 2005, Cuernavaca. Conferência Latino-Americana de Interação Humano-Computador. New York : ACM Press, 2005. v. 124. p. 295-300.

PANTON, K.; MATUSZEK, C.; LENAT, D. B.; SCHNEIDER, D.; WITBROCK, M. SIEGEL, N. SHEPARD, B. (2006). Common Sense Reasoning – From Cyc to Intelligent Assistant. In: **The National Conference On Innovative Applications Of Artificial Intelligence. Ambient Intelligence in Everyday Life**. New York (US): Springer, Lecture Notes in Artificial Intelligence, v. 3864, 2006. pp. 1-31.

PEARL, J. **Heuristics**: Intelligent Search Strategies for Computer Problem Solving, Addison-Wesley, Reading, MA., 1984.

PREECE, S. E. A spreading activation network model for information retrieval, University of Illinois at **Urbana-Champaign**, Champaign, IL, 1981

RICHENS, R. H. 1958. Interlingual machine translation. *Computer Journal*, 1(3):144--147.

SCORM - Sharable Content Object Reference Model Version 1.2 – The Scorm Content Aggregation Model; Advanced Distributed Learning, 2001. Disponível em: <[www.adlnet.org](http://www.adlnet.org)>. Acesso em mai. 2005.

SHELLEY, C. **Multiple analogies in Science and Philosophy**. Amsterdam/Philadelphia: John Benjamins Publishing Company, 2003.

SINGH, P. The public acquisition of commonsense knowledge. In: AAAI SPRING SYMPOSIUM ON ACQUIRING (AND USING) LINGUISTIC (AND WORLD) KNOWLEDGE FOR INFORMATION ACCESS. **Proceedings...** US (Palo Alto, CA): 2002.

SINGH, P. The OpenMind Commonsense project. **KurzweilAI.net**, 2002. Disponível em: <<http://web.media.mit.edu/~push/OMCSProject.pdf>>. Acesso em: fev. 2006.

STAMM, S. Mixed nuts: atypical classroom techniques for computer science courses. 2003. **Crossroads**, v. 10, n. 4. ACM Press.

TAYLOR, M. E.; MATUSZEK, C.; KLIMT, B.; WITBROCK, M. Autonomous Classification of Knowledge into an Ontology. In: TWENTIETH INTERNATIONAL FLAIRS CONFERENCE. **Proceedings...** Key West, FL, May 2007.

TALARICO NETO, A. ; ANACLETO, J. C. ; NERIS, V. P. A. . Linguagem de Padrões para Apoiar o Projeto de Material Instrucional para e-learning. In: XVI Simpósio Brasileiro de Informática na Educação - SBIE, 2005, Juíz de fora. **Anais do XVI Simpósio Brasileiro de Informática na Educação 2005**. Porto Alegre RS : Sociedade Brasileira de Computação, 2005. v. 1.

TALARICO, A. ; ANACLETO, J. C. ; NERIS, V. P. A. ; GODOI, M. S. ; CARVALHO, A. F. . Cognitor: um Framework baseado na Linguagem de Padrões Cog-Learn. In: Simpósio Brasileiro de Informática na Educação, 2006, Brasília. **Anais do XVI Simpósio Brasileiro de Informática na Educação**, 2006. v. 1. p. 529-538.

TURING, A. M. **Computing machinery and intelligence**. *Mind*, v.49, p.433-460, 1950. Disponível em: <<http://www.cs.swarthmore.edu/~dylan/Turing.html>>. Acesso em: jun. 2007.

WEST, C. K. *et al.* **Instructional Design: Implications from Cognitive Science**. Massachusetts (Boston): Allyn and Bacon, 1991.

ZHANG D. *et al.*. Can e-learning replace classroom learning? **Communications of the ACM**, US (New York, NY): ACM Press, v. 47, n. 5, mai. 2004. p. 75-79.