

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

**“Uso do Senso Comum na detecção de diferenças culturais no
contexto do Projeto OpenMind CommonSense no Brasil”**

Marie Tsutsumi

**São Carlos
Março/2006**

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

T882us

Tsutsumi, Marie.

Uso do Senso Comum na detecção de diferenças culturais no contexto do Projeto OpenMind CommonSense no Brasil / Marie Tsutsumi. -- São Carlos : UFSCar, 2007. 113 f.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2006.

1. Interação homem - máquina. 2. Senso comum. 3. OMCS (Open Mind Common Sense). 4. Rede semântica. 5. Diferenças culturais. I. Título.

CDD: 004.019 (20ª)

Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia

Programa de Pós-Graduação em Ciência da Computação

“Uso do Senso Comum na detecção de diferenças culturais no contexto do Projeto OpenMind CommonSense no Brasil”


MARIE TSUTSUMI

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:



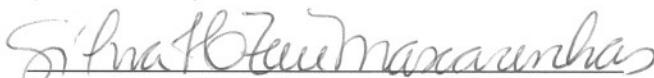
Profa. Dra. Júnia Coutinho Anacleto
(Orientadora – DC/UFSCar)



Profa. Dra. Heloisa Vieira da Rocha
(IC/UNICAMP)



Profa. Dra. Renata Pontin de Mattos Fortes
(ICMC/USP)



Profa. Dra. Silvia Helena Zem-Mascarenhas
(Denf/UFSCar)

*“Aceitar as diferenças e as opções de
cada pessoa garante uma convivência
harmoniosa e feliz”.*

Autor desconhecido

AGRADECIMENTOS

À minha orientadora Júnia Coutinho Anacleto pela confiança, pelo incentivo, pelo carinho e principalmente por sempre querer ver o melhor de mim.

Ao Walter pela paciência, pelo apoio e por sempre querer me ver feliz.

Ao meu irmão Marcos, à mamãe e ao papai que mesmo longe, transmitiram a segurança e a tranquilidade que precisava.

Aos amigos do coração Sullivan e Marina, sempre prontos a ajudar.

Aos colegas do LIA que me receberam de braços abertos e fizeram realmente me sentir parte do grupo. Ao André e à Vânia por transmitirem suas experiências. Ao Muriel, ao Fabiano e ao Edimilson que compartilharam as alegrias as dificuldades no dia a dia do trabalho no laboratório.

À Cris e ao Marcos, pela companhia nas horas do almoço.

À professora Renata Pontin de Mattos Fortes que desde os primeiros anos da graduação, de algum modo, acompanhou o meu trabalho.

À CAPES pela bolsa concedida no período de Setembro de 2004 a Março de 2006.

Essas palavras não são o suficiente para expressar a minha enorme gratidão. Tenham certeza que terei ótimas lembranças dessa etapa da minha vida graças a vocês.

RESUMO

Conhecer características específicas de cada cultura pode ser de grande utilidade na realização da comunicação intercultural, pois culturas diferentes têm noções distintas de como começar e como desenvolver uma conversa.

Com o intuito de ajudar as pessoas a lidar com essas diferenças em relação às normas culturais, têm-se trabalhado para que os computadores disponibilizem tal conhecimento, já que um ambiente computacional é um recurso bastante utilizado na comunicação entre pessoas distantes e de culturas diferentes.

Para que os computadores possam fornecer esse tipo de informação, têm sido coletadas grandes bases de conhecimento para representar o conhecimento de senso comum, pois, acredita-se que esse conhecimento de senso comum pode dar algumas diretrizes sobre as diferenças.

Neste estudo, foram utilizadas algumas bases de conhecimento contendo sentenças que descrevem o senso comum das pessoas e suas atividades diárias, crenças e valores, obtidas através da *Web* de colaboradores voluntários.

Além disso, são descritas experiências que utilizam essas bases de conhecimento, e são propostas aplicações computacionais que fazem buscas de características culturais relacionados aos hábitos alimentares para alertar os usuários sobre as possíveis diferenças entre as culturas. Desta forma, espera-se que o trabalho contribua para que os softwares possam analisar tais diferenças e também estimular a colaboração intercultural.

ABSTRACT

Knowing what is specific about each culture would be helpful to perform cross cultural communication, because different cultures have different notions of how to begin and develop conversations.

Due to these difficulties in differences in cultural norms, we are working to provide to computers this knowledge, since computer environment can allow communication across distance and across cultures.

In order to provide to computers cross cultural knowledge, we have collected large knowledge bases representing Common Sense knowledge because we believe that Common Sense knowledge can give some guidelines about different cultures.

In our research, we have used bases containing sentences describing common sense knowledge of people and everyday activities, beliefs and values collected from volunteer Web contributors.

Furthermore, we describe experiences with these knowledge bases, and we are proposing software which automatically searches for cultural differences related to eating habits, alerting the user to potential differences among cultures. In this way, we hope that our work will contribute to software that takes better account of such differences, and fosters inter-cultural collaboration

ÍNDICE DE FIGURAS E GRÁFICOS

Figura 1 – Arquitetura do Projeto Open Mind Common Sense	9
Figura 2 – Aquisição e armazenamento de fatos de senso comum	9
Figura 3 – Site Open Mind Common Sense	10
Figura 4 – Gerador da Conceptnet – conversão de sentenças em nós	10
Figura 5 – Funcionamento da API.....	11
Figura 6 – Parte da ConceptNet, a rede semântica do projeto.....	16
Figura 7 – Atividade com a forma livre.....	21
Figura 8 – Tratamento de nós duplicados.....	22
Figura 9 – Geração de nós que mostram propriedade	22
Figura 10 – Generalizações de nós	22
Figura 11 – Exibição dos termos relacionados ao contexto	24
Figura 12 – Analogias	25
Figura 13 – Arquitetura do projeto OMCS no Brasil.....	28
Figura 14 – Aquisição e armazenamento de fatos de senso comum em português.....	29
Figura 15 – Sentenças em forma de <i>templates</i>	29
Figura 16 – Gerador da ConceptNet brasileira	30
Figura 17 – Funcionamento da API brasileira.....	31
Figura 18 – Exemplo de retro-alimentação	40
Figura 19 – Esquema de retro-alimentação da base de dados OMCS no Brasil	42
Figura 20 – Tela com opção de retro-alimentação	42
Figura 21 – Trecho da ConceptNet brasileira	44
Figura 22 – Comunicação entre o Curupira e as funções de manipulação da ConceptNet.....	49
Figura 23 – Normalizador	50
Figura 24 – Trecho do dicionário utilizado pelo Normalizador	51
Figura 25 – Tratamento de nós duplicados Definição do valor da frequência.....	55
Figura 26 – Generalizações de nós	55
Figura 27 – Inexistência da relação IsA	55
Figura 28 – Alto valor do atributo “Poder” – site de uma universidade da Malásia	66
Figura 29 – Baixo valor do atributo “Poder” – site de uma universidade da Holanda	66
Figura 30 – What is He Thinking (WIHT)	76
Figura 31 – A versão brasileira do WIHT, com novas funcionalidades	78
Figura 32 – ConceptNets de cada cultura geradas a partir dos perfis dos colaboradores.....	79
Figura 33 – Fatos relacionados à comida e hábitos alimentares no Brasil	82
Figura 34 – Resultado de buscas nos sites do Brasil(A), EUA(B) e México (C)	84
Gráfico 1 – Porcentagem de cada faixa etária dos colaboradores	35
Gráfico 2 – Porcentagem dos níveis de escolaridade dos colaboradores.....	36
Gráfico 3 – Porcentagem dos gêneros dos colaboradores.....	37
Gráfico 4 – Ilustração da porcentagem dos estados de origem dos colaboradores.....	38

ÍNDICE DE TABELAS

Tabela 1 – Exemplos de elementos encontrados nos nós da ConceptNet	16
Tabela 2 – Relações da ConceptNet	17
Tabela 3 – Mais detalhes dos perfis dos usuários no site brasileiro.....	33
Tabela 4 – Faixa etária dos colaboradores.....	35
Tabela 5 – Escolaridade dos colaboradores.....	36
Tabela 6 – Gênero dos colaboradores.....	37
Tabela 7 – Estado de origem dos colaboradores	37
Tabela 10 – Tipos de conceitos ou nós	43
Tabela 11 – Declarações geradas diretamente a partir de <i>templates</i>	52
Tabela 12 – Horário das refeições	83
Tabela 13 – Tipos de comida	83
Tabela 14 – O que as pessoas comem em cada refeição?.....	85
Tabela 15 – Comidas e bebidas para ocasiões especiais.....	86
Tabela 16 – Comidas mencionadas pelos brasileiros.....	87

Lista de Abreviaturas

API – *Application Programming Interface*

IA – Inteligência Artificial

IHC – Interação Humano-Computador

IHH – Interação Humano-Humano

JSP – *Java Server Pages*

OMCS - Open Mind Common Sense

RI – Recuperação de Informação

WIHT – *What is he thinking*

ÍNDICE

Capítulo 1 - Introdução	1
1.1. Motivação	1
1.2. Objetivos.....	2
1.3. Organização do Trabalho.....	2
Capítulo 2 - Projeto Open Mind Common Sense	5
2.1. Considerações Iniciais	5
2.2. Conceito de Senso Comum	6
2.3. O Projeto Open Mind Common sense do MIT	7
2.3.1. Arquitetura do Projeto Open Mind Common Sense.....	8
2.3.2. O Site Open Mind Common Sense	12
2.3.3. A ConceptNet.....	15
2.3.4. MontyLingua.....	18
2.3.5. Gerador da ConceptNet.....	20
2.3.6. A API do Projeto Open Mind Common Sense.....	23
2.4. Considerações finais	25
Capítulo 3 - O Projeto Open Mind Common sense no Brasil	27
3.1. Considerações iniciais.....	27
3.2. Arquitetura do projeto Open Mind Common Sense do Brasil.....	28
3.3. O Site Open Mind Common Sense no Brasil.....	33
3.4. A ConceptNet	43
3.5. O Curupira.....	45
3.6. Gerador da ConceptNet	51
3.7. A API brasileira do Projeto Open Mind Common Sense	56
3.8. Considerações Finais.....	57
Capítulo 4 - Senso Comum para Reconhecimento de Diferenças Culturais	59
4.1. Considerações Iniciais	59
4.2. Importância do Conhecimento sobre Cultura em aplicações computacionais	61
4.3. Considerações Finais.....	69
Capítulo 5 - Usando Senso Comum para Facilitar a Comunicação entre Pessoas mostrando as Diferenças Culturais.....	71
5.1. Considerações Iniciais	71
5.2. Estudo de Caso – Análise dos hábitos alimentares.....	74
5.3. Considerações Finais.....	80
Capítulo 6 - Resultados e Discussão	81
6.1. Considerações Iniciais	81

6.2. Discussão do Estudo de caso - Senso Comum e os Hábitos Alimentares	81
6.3. Justificativa	87
6.4. Considerações Finais	88
Capítulo 7 - Conclusões	89
7.1. Considerações iniciais.....	89
7.2. Síntese dos Principais Resultados.....	89
7.3. Análise Crítica	90
7.4. Dificuldades a serem superadas	91
7.5. Publicações	92
7.6. Trabalhos futuros	93
Referências	95
Anexos.....	103

CAPÍTULO 1 - INTRODUÇÃO

1.1. Motivação

As máquinas ainda não são ferramentas muito espertas, pois não são capazes de se adaptarem a novas circunstâncias e nem fazer suposições óbvias sobre algo que o usuário está querendo fazer.

Na tentativa de construir máquinas que possam entender os objetivos dos usuários e que ajudem a alcançá-los, acredita-se que o senso comum (fatos que a maioria das pessoas aceitam como verdade) possa ajudar os computadores nessa tarefa.

Entretanto, para obter o conhecimento necessário para representar o senso comum de uma pessoa, é preciso alguns milhões de fatos (30 a 60 milhões, segundo Minsky (1986)). Como uma solução encontrada para obter todo esse conhecimento de senso comum, a equipe do Projeto Open Mind Common Sense (OMCS) está colhendo fatos de senso comum através da colaboração dos usuários da *Web*, já que todas as pessoas possuem esse tipo de conhecimento, que está relacionado aos fatos simples do dia-a-dia.

O Open Mind Common Sense (OMCS) é um projeto iniciado no MediaLab do Massachusetts Institute of Technology (MIT) com o qual o Laboratório de Interação Avançada (LIA) da UFSCar mantém parceria no desenvolvimento de uma versão brasileira do OMCS.

Percebendo que os fatos de senso comum poderiam ser utilizados para reconhecer diferenças culturais, verificou-se a possibilidade de utilizar o senso comum para amenizar algumas dificuldades existentes na comunicação entre pessoas de culturas diferentes.

1.2. Objetivos

Este trabalho propõe explorar as possibilidades do uso de conhecimento de senso comum para reconhecer algumas diferenças culturais. Entendendo as características específicas das diversas culturas, acredita-se que programas computacionais mais adequados para cada uma dessas culturas podem ser construídos.

Espera-se assim, que com a identificação das diferenças culturais, as aplicações possam facilitar a comunicação entre as pessoas de culturas diferentes. Além de identificar diferenças culturais entre países, o senso comum será utilizado para identificar diferenças culturais entre pessoas de idades, sexos, cidades, estados e graus de escolaridade diferentes, mostrando assim, o potencial do uso de senso comum para identificar características em todos os tipos de culturas¹.

O propósito principal deste trabalho é estudar os fatos de senso comum no Brasil para realizar análises comparativas entre os fatos inseridos por colaboradores de outros países e também analisar os fatos fornecidos por colaboradores brasileiros considerando seus perfis, com o intuito de mostrar as diferenças existentes entre os grupos culturais. Deste modo é possível fornecer uma base para viabilizar o uso desse conhecimento em aplicações computacionais.

1.3. Organização do Trabalho

Este trabalho encontra-se organizado da seguinte maneira: no capítulo 2 serão abordadas questões relativas ao conceito de senso comum e será apresentado o Projeto Open Mind Common Sense, que está tentando fornecer esse tipo de conhecimento aos computadores. A seguir, será mostrada a forma adotada para representar o senso comum computacionalmente através da utilização de uma rede semântica, a Conceptnet. Também será apresentado o modo para gerá-la e como utilizá-

¹ Cultura é um padrão de comportamento de um grupo (DICIONÁRIO AURÉLIO, 1995)

la em aplicações computacionais com o auxílio do analisador de língua inglesa, o MontyLingua.

No capítulo 3 será apresentado o projeto OMCS no Brasil, como o site e a rede semântica foram desenvolvidos neste trabalho e como o Curupira, analisador de sentenças da língua portuguesa desenvolvido pelo Núcleo Interinstitucional de Lingüística Computacional (NILC), (<http://www.nilc.icmc.usp.br/nilc/index.html>) apóia esse sistema. A seguir, será apresentada uma visão mais detalhada de cada parte do sistema OMCS, bem como as soluções para o processamento de linguagem natural em português desenvolvidas neste trabalho. Também serão citadas algumas pesquisas que têm sido desenvolvidas no Brasil com o uso do senso comum. Além disso, serão apresentadas as atividades disponíveis no site brasileiro e os perfis das pessoas inseriram fatos de senso comum. Será apresentado o funcionamento do esquema de retro-alimentação de dados para geração de novas sentenças. E serão mencionados os meios que foram utilizados para atrair e incentivar as pessoas a inserirem fatos de senso comum para colaborar com o projeto.

No capítulo 4 será apresentado como o conhecimento de senso comum pode ajudar na identificação de diferenças existentes entre as culturas e serão exemplificadas algumas aplicações computacionais que podem ser auxiliadas com o reconhecimento das características culturais dos usuários.

No capítulo 5 será discutida a importância de entender as diferenças culturais durante a comunicação entre as pessoas de culturas diferentes. Também serão apresentados alguns atributos e características de linguagem que podem influenciar na qualidade da comunicação dos usuários. Em seguida, será apresentado um protótipo construído neste trabalho que mostra como uma aplicação computacional pode utilizar o senso comum para mostrar as diferenças existentes entre as pessoas de diferentes países,

estados, cidades, faixas etárias, sexos, graus de instrução ou qualquer outra diferença cultural.

O capítulo 6 apresenta os resultados obtidos nas análises realizadas nas bases de senso comum coletados através dos sites OMCS.

Por fim, no capítulo 7, é apresentada a síntese dos principais resultados, a análise crítica, as dificuldades a serem superadas, as publicações e os trabalhos futuros.

CAPÍTULO 2 - PROJETO OPEN MIND COMMON SENSE

2.1. Considerações Iniciais

Apesar de um computador poder jogar xadrez muito bem, resolver complexos problemas de logística e ter grande sucesso em várias aplicações computacionais práticas, ele ainda não é capaz de entender uma simples história infantil (LIU E SINGH, 2004A). Um computador é capaz de realizar algumas tarefas com desempenho melhor que muitos especialistas de uma determinada área, mas ele ainda não é capaz de reconhecer objetos, escrever sentenças ou chegar a simples conclusões que uma criança de 3 anos consegue. Isso porque os computadores são capazes de realizar tarefas complexas em um domínio limitado. O problema é que os computadores não possuem “senso comum” (SINGH, 2002A), o conhecimento do cotidiano que toda pessoa aprende durante a vida, termo que será explicado melhor na próxima seção.

Com o objetivo de tentar construir computadores mais inteligentes, o projeto Open Mind Common Sense (SINGH, 2002A) tenta utilizar esse ‘senso comum’ para construir aplicações computacionais mais amigáveis aos usuários e que os ajudem a evitar cometer erros tolos (LIEBERMAN ET AL., 2004).

O desafio é como fornecer aos computadores o senso comum. Além disso, é preciso investigar como dar a capacidade de raciocínio aos computadores. Ter um banco de dados com alguns milhões de fatos não é suficiente. É preciso fornecer às máquinas os diferentes modos de utilizar o conhecimento, para que possam pensar, planejar, explicar, prever e todas as outras habilidades mentais de um humano (SINGH, 2002A).

Na próxima seção, será explicado com mais detalhe esse termo senso comum; na seção 2.3 serão apresentados o projeto Open Mind Common Sense e suas

abordagens para obtenção e utilização dos fatos de senso comum; por fim, na seção 2.4 serão apresentadas as considerações finais desse capítulo.

2.2. Conceito de Senso Comum

Marvin Minsky (MINSKY, 1986), um dos maiores pesquisadores da área de Inteligência Artificial (IA), conceitua senso comum da seguinte maneira:

Os conhecimentos de senso comum são as habilidades mentais que as pessoas compartilham. O pensamento de senso comum é muito mais complexo do que outras realizações intelectuais que atraem mais atenção e respeito, as quais chamamos de “perícia”. Estas envolvem grande quantidade de conhecimento, mas normalmente só empregam alguns tipos de representações, em contraste com o senso comum, que envolve muitos tipos de representações e por isso requer uma gama maior de habilidades (MINSKY, 1986). Tradução livre.

Assim, pode-se perceber que senso comum representa a imensa porção da experiência humana, que abrange conhecimento sobre aspectos especiais, físicos, sociais, temporais e psicológicos da vida cotidiana (LIU E SINGH, 2004A). Liu e Singh (2004A) dizem também que é um tipo de conhecimento bem genérico sobre o dia-a-dia e que todas as pessoas possuem em um determinado tempo e lugar, ou seja, em uma determinada cultura.

Enquanto as pessoas consideram senso comum como sinônimo de um bom julgamento, no contexto do projeto Open Mind Common Sense (OMCS) o termo senso comum é utilizado para se referir aos fatos que a maioria das pessoas aceitam como verdade. Por exemplo: o gelo é gelado; para beber uma xícara de café, primeiro deve-se

colocar café dentro de uma xícara; se você esquecer o aniversário de um amigo, ele pode ficar triste com você (SINGH, 2002A).

Segundo as definições anteriores, pode-se dizer que o senso comum nem sempre representa uma verdade absoluta, já que esse conceito se refere ao que as pessoas acreditam ser verdade. Assim, se a maioria das pessoas de um determinado local pensar que a terra não é redonda, esse fato é visto como senso comum entre elas, mesmo que não seja uma verdade.

Também é preciso mencionar que para representar todo senso comum que uma pessoa comum possui, é preciso uma grande quantidade de conhecimento (30 a 60 milhões de fatos, segundo Minsky (1986)) e de diversos tipos.

Desse modo, o senso comum pode ser usado para reduzir a necessidade do usuário explicitar toda a informação detalhadamente, pois a máquina pode fazer suposições sobre o que o usuário pode estar querendo numa situação específica. Um exemplo dos citados em (LIU E SINGH, 2004A) mostra que quando um usuário digita uma frase como *'I ate some chips with my lunch'*, a palavra *'chips'* provavelmente se refere a *'potato chips'*, e não a *'computer chips'*. Para isso não foi necessário explicitar detalhes do conceito de *'chips'* dizendo que se trata de uma comida, feita de batata, etc. O computador deveria entender o sentido da palavra pelo contexto que ela está sendo utilizada

2.3. O Projeto Open Mind Common sense do MIT

Esse sonho de criar computadores com conhecimento humano é um objetivo antigo, começou na década de 60. Porém, muitos cientistas abandonaram o projeto, não apenas pela dificuldade, mas também porque as pessoas preferem projetos de IA mais práticos como os da robótica e sistemas especialistas (DENISON, 2003).

Mais tarde, Douglas Lenat (1990) começou a construir uma base de dados de conhecimento de senso comum, o projeto Cyc. Muitas pessoas o consideram um herói por persistir nessa pesquisa mesmo com inúmeras críticas. Lenat e sua equipe trabalham desde 1984, e já gastaram dezenas de milhões de dólares, inserindo fatos manualmente para construir a sua base (SINGH, 2002A).

Como a representação do senso comum requer um número muito grande de fatos (30 a 60 milhões de fatos, segundo Minsky (1986)), inspirado no sucesso dos projetos colaborativos através da web, a equipe do Projeto Open Mind Common Sense lançou um site (<http://openmind.media.mit.edu>), para construir sua base de conhecimento através da colaboração de voluntários pela Web ao invés de fazer isso manualmente, como nos projetos anteriores.

Além do site, iniciou-se também o desenvolvimento de um analisador de língua inglesa, o MontyLingua (MONTYLINGUA, 2005), de uma rede semântica, a ConceptNet (LIU E SINGH, 2004A; SINGH ET AL., 2004; LIU E SINGH, 2004B), e de uma API (*Application Programming Interface*) com procedimentos para manipulação da rede (LIU E SINGH, 2004B), com o intuito de viabilizar a utilização dos fatos de senso comum armazenados na base de dados em aplicativos computacionais.

2.3.1. Arquitetura do Projeto Open Mind Common Sense

A figura 1 apresenta esquematicamente a arquitetura do projeto Open Mind Common Sense, para uma visão geral.

A seguir, cada um dos elementos da arquitetura do projeto serão explicados com mais detalhes:

Site: é utilizado para colher fatos de senso comum. Ele apresenta uma sentença para ser preenchida pelo usuário, por exemplo, *'An object that you might find*

in a restaurant is ___’ ou em forma livre. O dado inserido é armazenado num **banco de dados**, aumentando assim, a base de conhecimento (seta 1 da figura 2). Os dados armazenados são utilizados pelo sistema para formar novas sentenças (seta 2 da figura 2).

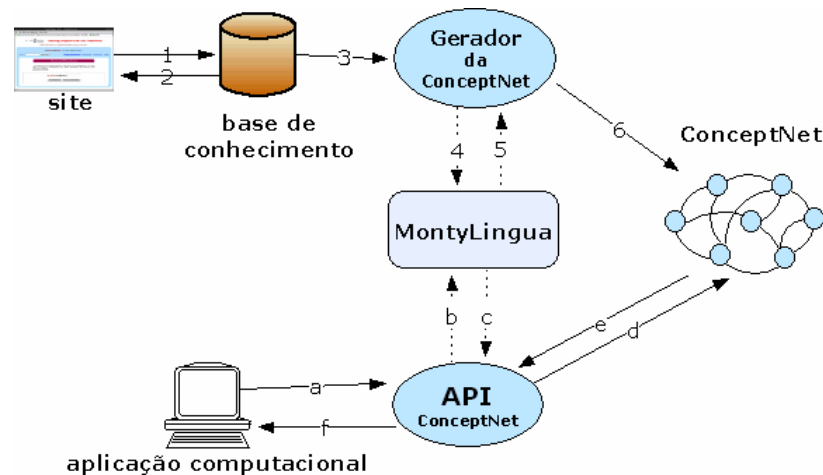


Figura 1 – Arquitetura do Projeto Open Mind Common Sense

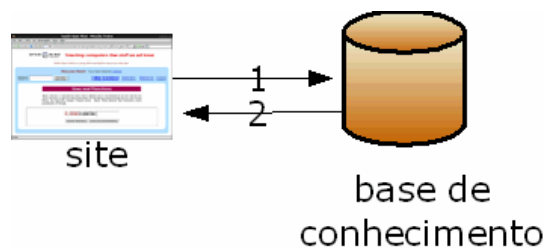


Figura 2 – Aquisição e armazenamento de fatos de senso comum

Assim, quando o usuário preenche uma sentença “*An object that you might find in a restaurant is_*” com “*fried potato*”, novas sentenças com o termo “*fried potato*” serão formadas como “*You generally find a **fried potato** in a ___*”, como mostrado na figura 3.

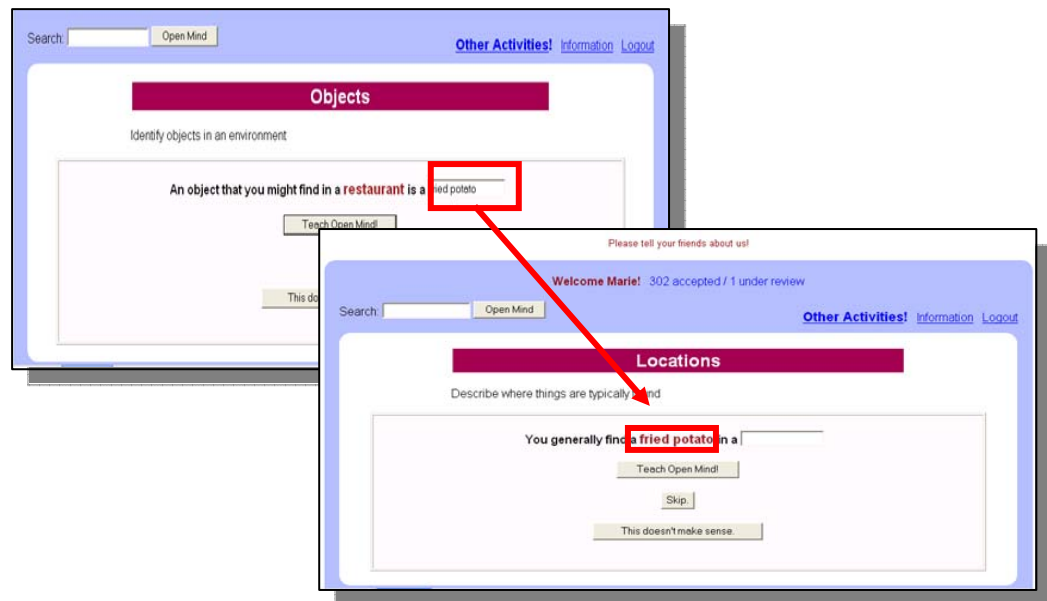


Figura 3 – Site Open Mind Common Sense

Gerador da ConceptNet: As sentenças inseridas pelos usuários armazenadas na base de conhecimento são transformadas em nós de uma rede semântica, a **ConceptNet**.

Exemplo de uma sentença transformada em um nó:

“An object that you might find in a restaurant is fried potato” → *(LocationOf, fried potato, restaurant)*

O gerador da ConceptNet utiliza uma sentença recuperada da base de conhecimento (seta 3 da figura 4). Com o auxílio de um analisador de língua inglesa, o **MontyLingua** (setas 4 e 5 da figura 4), o gerador converte a sentença em nó (seta 6 da figura 4), que é um formato em que os computadores podem utilizar.

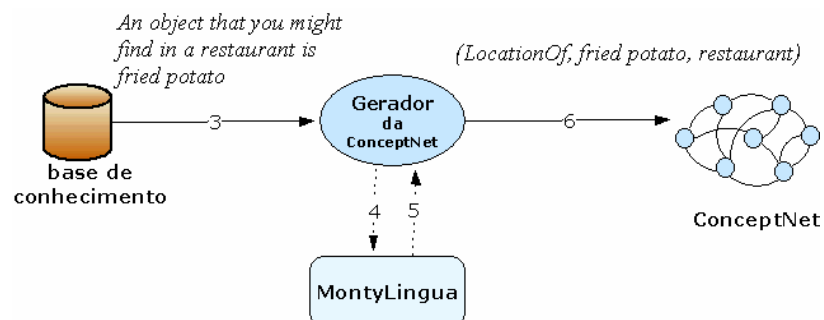


Figura 4 – Gerador da Conceptnet – conversão de sentenças em nós

API: O funcionamento da API é descrito em 5 passos:

1. Inicialmente o usuário da aplicação computacional digita um texto, por exemplo, “Jantaremos em casa?” (seta “a” da figura 5).

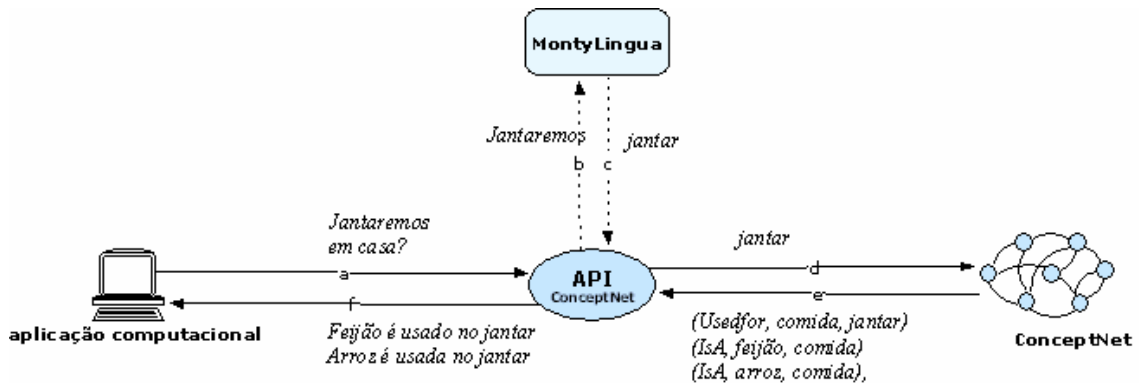


Figura 5 – Funcionamento da API

2. Então, a API chama métodos do MontyLingua, que faz a análise do texto digitado (seta “b” da figura 5), e retorna o texto um formato adequado (seta “c” da figura 5) para que a API possa utilizar. Por exemplo, depois da análise da palavra “Jantaremos”, o MontyLingua retorna a forma normal de “Jantaremos” que é “jantar” e envia de volta para a API.

3. A API utiliza esse termo, agora na forma canônica, como a expressão de busca na rede semântica (seta “d” da figura 5).

4. Depois, a API obtém os nós relacionados ao termo considerando o contexto, (seta “e” da figura 5).

5. Então a API envia os nós em um formato legível ao usuário (seta “f” da figura 5). Depois, a aplicação computacional mostra ao usuário, textos com algumas sugestões relacionadas ao texto digitado pelo usuário inicialmente.

Vale mencionar que a aplicação computacional nem sempre fornece uma resposta completa ou exata de um problema. Algumas vezes apresenta sugestões

relevantes que podem facilitar o trabalho do usuário (denominadas Aplicações '*fail soft*²').

2.3.2. O Site Open Mind Common Sense

Um modo que a IA tem utilizado para representar o conhecimento é através de fatos³. Essa representação foi inicialmente utilizada no projeto Cyc (LENAT ET AL., 1990) quando se tentou formalizar o senso comum, em uma representação lógica. Porém, apenas os engenheiros da empresa responsável pela Cyc, a Cycorp, são capazes de transformar um texto em língua natural nessa representação, pois essa representação lógica é escrita numa linguagem específica, a CycL. Manualmente, até 2004, os engenheiros haviam inserido 1,6 milhões de fatos na base de conhecimento (SINGH ET AL., 2004).

Porém, adquirir uma base de conhecimento que engloba toda quantidade de fatos necessários para representar todo o senso comum que uma pessoa possui e representar toda a variedade de conhecimento que esse tipo de conhecimento engloba, não é uma tarefa muito simples, de acordo com Singh (2004).

A respeito da quantidade de fatos necessários, um dos grandes pesquisadores da IA, Marvin Minsky (MINSKY, 1986), uma vez estimou que “senso comum é um conhecimento de 30 a 60 milhões de coisas⁴ sobre o mundo”. Em relação à variedade de conhecimento, é preciso representar vários tipos de coisas que as pessoas podem pensar, como por exemplo assuntos sociais, econômicos, políticos, psicológicos, matemáticos, etc. Devido a esses fatores, inserir todo o conhecimento de senso comum parece não ser uma tarefa muito simples de se fazer manualmente.

² Aplicações *fail Soft*: Aplicações computacionais que mostram resposta nem sempre corretas (LIEBERMAN E SCHMANDT, 2005)

³ Fatos: Refere-se às sentenças expressas em língua natural

⁴ Coisas: refere-se a fatos

Já que todas as pessoas têm o senso comum que o computador precisa, então envolvê-las na construção da base de conhecimento que é necessário dar aos computadores seria uma iniciativa interessante. São tantas pequenas coisas que a colaboração das pessoas poderia ajudar a criar um banco de dados que pudesse dar a visão da realidade. A partir dessa idéia, Push Singh criou o site Open Mind Common Sense (<http://openmind.media.mit.edu>), inspirado no sucesso dos projetos distribuídos e colaborativos através da Web. Assim, o projeto passou a contar com a colaboração do público geral, que pode ajudar escrevendo algumas sentenças em língua natural (LIU E SINGH, 2004A; SINGH ET AL., 2004).

Essa é a grande diferença entre o OMCS as grandes bases de conhecimento, como o Cyc (LENAT ET AL., 1990) e WordNet (<http://wordnet.princeton.edu/>). Estas são geradas manualmente, enquanto a base de conhecimento do Open Mind Common Sense é gerada automaticamente a partir das sentenças obtidas pelo site (LIU E SINGH, 2004A).

O objetivo do Projeto OMCS é reunir um enorme banco de dados com conhecimentos sobre o mundo que nos cerca considerando que qualquer pessoa pode “ensinar” essas coisas. Para isso, qualquer pessoa pode entrar no site e participar do projeto. Basta se cadastrar e começar a responder às perguntas que surgem na tela – sempre sobre coisas que parecem muito simples para nós, mas são extremamente desconhecidas para qualquer computador.

O site foi lançado em setembro de 2000, possuindo mais de 20.000 pessoas contribuindo e mais 811.000 fatos inseridos⁵. Em agosto de 2005 foi lançada a versão brasileira (<http://lia.dc.ufscar.br/sensocomum>) atualmente com mais de 1000 colaboradores e quase 100.000 fatos inseridos⁶. Em janeiro de 2006, o site mexicano

⁵ Dados obtidos em 10/01/2006

⁶ Dados obtidos em 12/03/2006

(<http://openmind.fi-p.unam.mx>) foi disponibilizado, hoje com 14 colaboradores e 246 fatos inseridos⁷. Outros sites pelo mundo também estão em construção, como o japonês, o coreano e o chinês (ANACLETO ET AL., 2006B). Eles têm como objetivo, colher grande quantidade de fatos de senso comum dos mais diversos tipos de pessoas, em diversas culturas.

A vantagem, segundo Singh (2004), nessa abordagem é que as pessoas não têm que aprender uma nova linguagem para inserir senso comum, pois elas usam a linguagem que já conhecem.

Para que as pessoas comecem a contribuir, basta entrar no site, preencher um pequeno cadastro no site, fornecendo alguns dados pessoais e definindo um *login* e uma senha. Depois é só ir completando as sentenças apresentadas aleatoriamente, como por exemplo, “Um fogão é usado para ___”; “O efeito de dormir é ___”; “Algo que você pode encontrar em uma cozinha é um(a) ___”. O site também permite ao usuário escolher uma atividade específica para inserir dados. Atualmente, existem 29 atividades, as quais capturam informações como por exemplo: Que tipo de coisa uma coisa é? (Mesa é um móvel); Quais são os efeitos de uma ação? (Tomar chuva te deixa molhado); Que tipos de coisas as pessoas querem? (Pessoas gostam de estar sadias); Onde uma coisa é comumente encontrada? (Você freqüentemente encontra um fogão na cozinha); Para que um objeto serve? (Um fogão serve para cozinhar); Como você resolve problemas? (Se você quer estar sem fome, coma algo). Essas atividades refletem os padrões observados nas entradas de forma livre⁸ (LIU E SINGH, 2004A; LIU E SINGH, 2004B).

Com essas atividades, é possível colher alguns tipos de conhecimento, tais como (SINGH, 2002B):

⁷ Dados obtidos em 20/03/2006

⁸ Forma livre: O texto a ser digitado não possui uma estrutura previamente definida.

- Taxonomia: “Gatos são mamíferos”
- Causa – “Você sente menos fome após comer um lanche”
- Objetivo – “Pessoas se agasalham para se manter aquecidas”
- Espaço – “Você pode encontrar um sofá em uma sala”
- Função – “Uma cadeira é para sentar”
- Planejamento – “Para não chegar atrasado em um compromisso saia cedo de casa”

Com o intuito de estimular os usuários a manter contribuindo, uma lista com os nomes dos usuários que mais colaboraram é divulgada no site, uma forma encontrada para promover uma disputa entre eles, estimulando maior contribuição.

2.3.3. A ConceptNet

As sentenças obtidas do site do OMCS não são diretamente computáveis. Os fatos inseridos em língua natural são convertidos para uma forma que pode ser utilizada por um computador (LIU E SINGH, 2004A). Elas são transformadas em uma rede semântica, a ConceptNet.

A ConceptNet é uma rede semântica de larga escala de conhecimento de senso comum extraído automaticamente a partir dos dados adquiridos pelo site, integrada com uma ferramenta de processamento de língua natural que trabalha com os textos do mundo real (LIU E SINGH, 2004A). Na figura 6, uma pequena parte da rede semântica é apresentada.

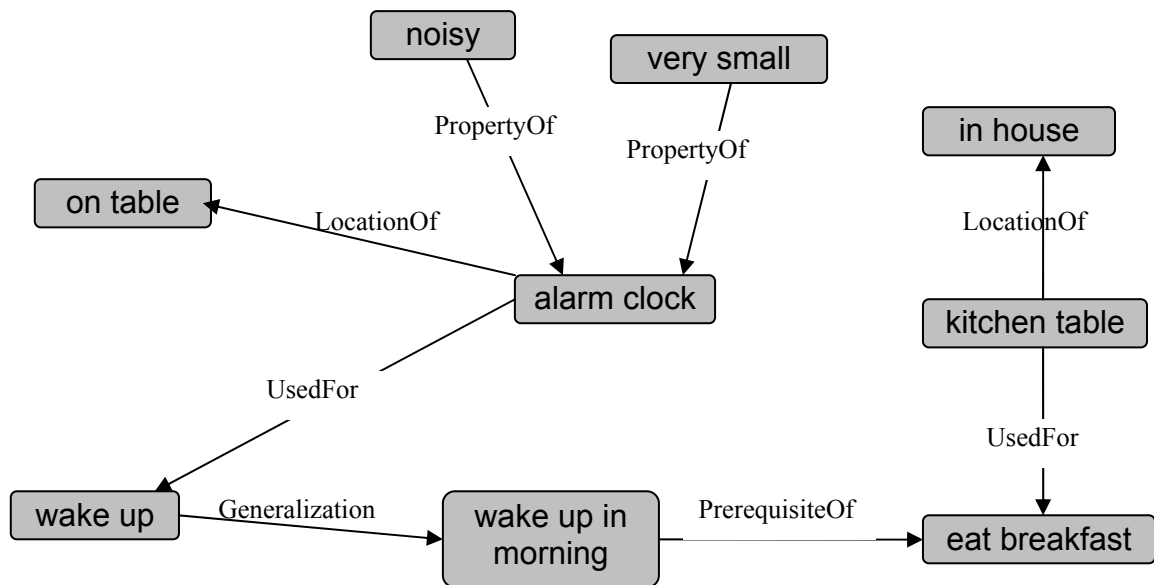


Figura 6 – Parte da ConceptNet, a rede semântica do projeto

A ConceptNet é formada por um conjunto de declarações⁹. Uma declaração consiste de conceitos e relações, mais a frequência e a inferência, como definidos a seguir:

- **Conceito:** pode ser um objeto, evento, propriedade ou lugar. Um conceito representa os nós na rede semântica (figura 6). Cada nó é um fragmento em inglês, como os exemplos mostrados na tabela 1.

Tabela 1 – Exemplos de elementos encontrados nos nós da ConceptNet

Verbo (Evento)	Frase nominal (Coisa)	Frase preposicional (Lugar)	Frase Adjetiva (Propriedade)
wake up	kitchen table	in house	very small
wake up in morning	alarm clock	on table	noisy
eat breakfast			

- **Relação:** é relação semântica que existe entre dois conceitos. A relação é representada por uma ligação entre os nós, como os exemplos “Generalization”, “UsedFor”, “PrerequisiteOf” e “LocationOf” da figura 6. Na ConceptNet existem

⁹ Declaração é o termo utilizado neste trabalho como tradução de assertion

atualmente 20 tipos de relações (coluna “Relação” da tabela 2) agrupadas em classes (coluna “Classe” da tabela 2). Essas 20 relações foram escolhidas porque elas apareciam com grande frequência na base OMCS. Elas refletem os padrões observados nas entradas de forma livre (LIU E SINGH, 2004A; LIU E SINGH, 2004B).

Tabela 2 – Relações da ConceptNet

CLASSE	RELAÇÃO	EXEMPLO
K-Lines	ConceptuallyRelatedTo	(ConceptuallyRelatedTo ‘bad breath’ ‘mint’ ‘f=4;i=0’)
	ThematicKLine	(ThematicKLine ‘wedding dress’ ‘veil’ ‘f=9;i=0’)
	SuperThematicKLine	(SuperThematicKLine ‘western civilisation’ ‘civilisation’ ‘f=0;i=12’)
Things	IsA	(IsA ‘horse’ ‘mamal’ ‘f=17; i=3’)
	PropertyOf	(PropertyOf ‘fire’ ‘dangerous’ ‘f=17;i=1’)
	PartOf	(PartOf ‘butterfly’ ‘wing’ ‘f=5;i=1’)
	MadeOf	(MadeOf ‘bacon’ ‘pig’ ‘f=3;i=0’)
	DefinedAs	(DefinedAs ‘meat’ ‘flesh of animal’ ‘f=2;i=1’)
Agents	CapableOf	(CapableOf ‘dentist’ ‘pull tooth’ ‘f=4;i=0’)
Events	PrerequisiteEventOf	(PrerequisiteEventOf ‘read letter’ ‘open envelope’ ‘f=2;i=0’)
	FirstSubeventOf	(FirstSubeventOf ‘start fire’ ‘light math’ ‘f=2;i=3’)
	SubEventOf	(SubEventOf ‘play sport’ ‘score goal’ ‘f=2;i=0’)
	LastSubeventOf	(LastSubeventOf ‘attend classical concert’ ‘applaud’ ‘f=2;i=1’)
Spatial	LocationOf	(LocationOf ‘army’ ‘in war’ ‘f=3; i=0’)
Causal	EffectOf	(EffectOf ‘view video’ ‘entertainment’ ‘f=2;i=0’)
	DesirousEffectOf	(DesirousEffectOf ‘sweat’ ‘take a shower’ ‘f=3;i=1’)
Functional	UsedFor	(UsedFor ‘fire place’ ‘burn’ ‘f=1;i=2’)
	CapableOfReceivingAcion	(CapableOfReceivingAcion ‘drink’ ‘serve’ ‘f=0; i=14’)
Affective	MotivationOf	(MotivationOf ‘play game’ ‘compete’ ‘f=3;i=0’)
	DesireOf	(DesireOf ‘person’ ‘not be depressed’ ‘f=2;i=0’)

- **Frequência (f):** é o número de vezes que a declaração aparece em toda base do OMCS. Por exemplo: a declaração (DesireOf ‘pessoa’ ‘não ser triste’ ‘f=2;i=0’), tem o valor da frequência igual a 2 (f=2), isso significa que existem duas sentenças na base OMCS dizendo que o desejo de uma pessoa é não ser triste.

- **Inferência (i):** é o número de declarações que foram necessárias para criar a nova declaração. Por exemplo, da sentença ‘Um pardal é um pássaro’, é extraída a declaração (IsA, ‘pardal’, ‘pássaro’ ‘f=1; i=0’). Como esta declaração foi gerada diretamente de uma sentença, não foi necessária nenhuma outra declaração para criá-la,

então valor da inferência é zero ($i=0$). A partir dela, pode ser inferida uma nova declaração como (CapableOf, 'pardal' 'voar' 'f=1; i=1'). A partir da declaração “Um pardal é um pássaro” foi necessária utilizar uma outra declaração “Um pássaro é capaz de voar” para gerar a declaração “Um pardal é capaz de voar”, por isso, o valor da inferência é 1 ($i=1$).

Essas declarações são geradas a partir da base do OMCS com o auxílio de uma ferramenta de processamento de língua natural, o MontyLingua.

2.3.4. MontyLingua

O MontyLingua é um analisador de língua inglesa escrito em linguagem Python e também disponível em linguagem Java. Dado um texto, ele mostra sua interpretação semântica. O MontyLingua extrai o sujeito, verbo e o objeto das sentenças, além das frases adjetivas e nominais verbais. Também recupera nomes, lugares, eventos, datas e outras informações semânticas.

Para isso, o MontyLingua possui classes para realizar as tarefas (descritas a seguir) sobre um texto:

MontyTokenizer: Separa o texto em “*tokens*”. Cada *token* representa uma palavra. Essa separação é feita separando as palavras pelos espaços em branco e caracteres especiais como ponto, vírgula, etc. Por exemplo, a sentença “*My dogs are very small*”, seria separada em “*My*”, “*dogs*”, “*are*”, “*very*” e “*small*”. Quando aparecem contrações do tipo “*you’re*”, isto é transformado em “*you are*”, e depois separado em “*you*” e “*are*”.

MontyTagger: Uma vez *tokenizado*, são colocadas as *tags* para indicar o que é sujeito, verbo, objeto, artigo, preposição, etc. Para isso é utilizado o conjunto de *tags* Penn Treebank (Ver Anexo A). Por exemplo: “*My/PRP\$*”, “*dogs/NNS*”,

“*are/VBP*”, “*very/RB*” e “*small/JJ*” onde “*PRP\$*” significa um pronome possessivo, “*NNS*” um substantivo no plural, “*VBP*” um verbo o presente não em terceira pessoa, “*RB*” um advérbio e “*JJ*” um adjetivo (Ver tabela Penn Treebank no Anexo A).

MontyChunker: Depois de separar os textos em *tokens* e colocar *tags* neles, os *tokens* agora são agrupados em *chunks*. Um chunk pode ser:

- Frase nominal (NX): frase que possui um substantivo como núcleo. Por exemplo: “*my dogs*”.
- Frase verbal (VX): frase que possui um verbo como núcleo. Por exemplo: “*are*”
- Frase adjetiva (AX): frase que possui um adjetivo como núcleo. Por exemplo: “*very small*”

A saída completa seria: “(NX) *My dogs* (NX)”, “(VX)*are* (VX)” e “(AX)*very small* (AX)”.

MontyLemmatizer: Coloca as sentenças na forma canônica, mudando verbos flexionados para a forma infinitiva e substantivos são passados para o singular: “*My dogs are very small*” transformaria em “*My dog is very small*”.

MontyNLGenerator: Gera um sumário de um texto, porém nem sempre apresenta resultados adequados. Por exemplo: O MontyNLGenerator gerou para a sentença “*My dog are very small*” o texto “*My dog was small*”, mudando o sentido da frase. Testando frases mais longas, por exemplo, “*Providing Interaction among Culturally Different People Using Common Sense*” o programa retornou o seguinte resultado: “*Provide interaction. Different people used common sense*”, que também mudou o sentido da frase original.

Visto o que o MontyLingua faz, agora será descrito onde e como ele ajuda no processo de geração da ConceptNet e no uso da ConceptNet por uma aplicação computacional.

2.3.5. Gerador da ConceptNet

A ConceptNet é gerada a partir de um processo automático, através da extração das sentenças semi-estruturadas¹⁰ da base do OMCS (LIU E SINGH, 2004A).

Esse processo ocorre em 3 fases:

Fase de extração

As sentenças extraídas através do site OMCS, armazenadas na base, são transformadas em declarações.

Aproximadamente 50 regras de extração transformam as sentenças em declarações da ConceptNet de forma direta. Isso é possível graças às atividades do site OMCS com formato semi-estruturado (*templates*), atividades em que o colaborador deve apenas preencher um campo de um formulário.

Exemplo:

O efeito de andar na chuva é ___ → (EffectOf, 'andar na chuva', '___', 'f=1; i=0')

Já as atividades de texto livre precisam passar por uma análise, pois não se sabe o que o usuário vai escrever, como mostra a figura 7.

¹⁰ Semi-estruturado: Neste trabalho, o termo “semi-estruturado” é usado para designar uma sentença com formato fixo mais um campo para o usuário completar. Por exemplo, “Uma mesa é usada para ___”.



Figura 7 – Atividade com a forma livre

O MontyLingua, responsável por essa análise, separa o texto em sujeito, verbo, objeto, artigo, preposição, etc. As sentenças sofrem uma análise sintática e semântica.

A relação formada com a forma livre é a k-line ‘ConceptuallyRelatedTo’, que é mais genérico que as demais, já que não se espera uma entrada com um formato específico.

Fase de normalização

As declarações extraídas na fase de extração são normalizadas. Primeiro, o MontyLingua classifica cada nó e cada palavra sintaticamente, isto é, classifica em verbo, sujeito, etc. Então, são verificados se os nós estão em um dos tipos de categorias: verbo, frase nominal, frase preposicional ou frase adjetiva. Feito isso, os artigos e outros determinadores (*the, a, two*) e os *modals* (*might, could, may*) são retirados. E por fim, as palavras são reduzidas à forma canônica (*lemma*) em relação às flexões e quanto ao número .

Exemplos:

‘is/are/were’ → *‘be’*

apples → *apple*

Fase de relaxamento

Depois de obter as declarações normalizadas, as declarações passam por um conjunto de heurísticas de relaxamento. A seguir é apresentada uma breve visão sobre as heurísticas:

1. As declarações duplicadas são retiradas e o número de repetições que existia é contabilizado no campo “frequência”. Assim, como mostrado no exemplo da figura 8, se havia três declarações, passará a existir uma só, com frequência 3.

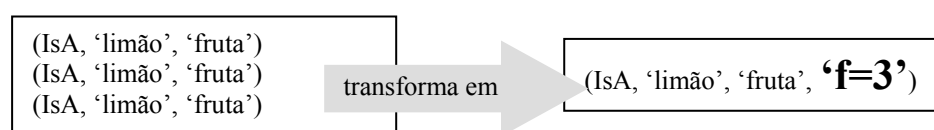


Figura 8 – Tratamento de nós duplicados

2. As declarações com a relação ‘IsA’ são usadas para gerar novas outras. O exemplo da figura 9 mostra que se maçã, banana e pêra são frutas, e cada uma delas são doces, implica que fruta tem a propriedade de ser doce.

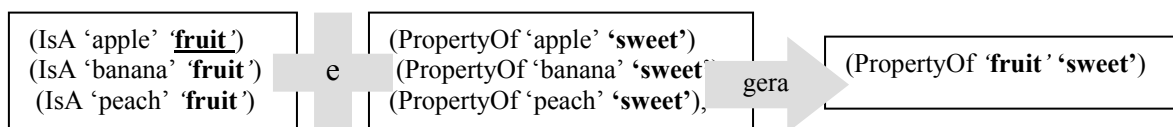


Figura 9 – Geração de nós que mostram propriedade

3. Generalizações temáticas e léxicas são produzidas relacionando o conhecimento mais específico ao mais geral, produzindo assim, relações do tipo “SuperThematicKLine”. O exemplo da figura 10 mostra que ‘comprar’ é uma forma genérica de ‘comprar comida’ e ‘adquirir comida’.

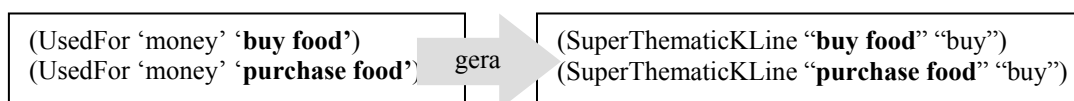


Figura 10 – Generalizações de nós

4. Diferenças de vocabulário como “bike” e “bicycle”, variações morfológicas como “relax” e “relaxation” são tratadas, através da adição de uma relação do tipo “SuperThematicKLine”.

5. Todas as declarações criadas têm os valores das inferências calculados.

2.3.6. A API do Projeto Open Mind Common Sense

A API desenvolvida no projeto Open Mind Common Sense é responsável por ajudar as máquinas a utilizar o conhecimento de diferentes modos, para que possam pensar, planejar, explicar, prever e todas as outras habilidades mentais de um humano.

Na prática ela faz a comunicação entre a aplicação computacional com a rede semântica. Quando o usuário da aplicação computacional digita um texto, é feita uma análise pelo MontyLingua e algumas funções da API são chamadas. Através da API, três tipos de inferência na ConceptNet podem ser realizados:

1. Encontrar o contexto

A função `GetContext()` determina o contexto sobre um conceito ou sobre vários conceitos. Por exemplo, quando um usuário digita o fragmento de uma sentença “ir para cama”, a rede semântica retorna os conceitos relacionados que foram encontrados, tais como, “despir”, “ir dormir”, “dormir”, “deitar-se”, “fechar os olhos”, “desligar a luz”, “sonho”, “escovar os dentes”, “roncar”, etc. Esta lista de nós que o `GetContext()` retorna está relacionada à distância entre nós e o número de caminhos. A figura 11, uma interface gráfica mostra os resultados obtidos com o uso da função `GetContext()` da API.

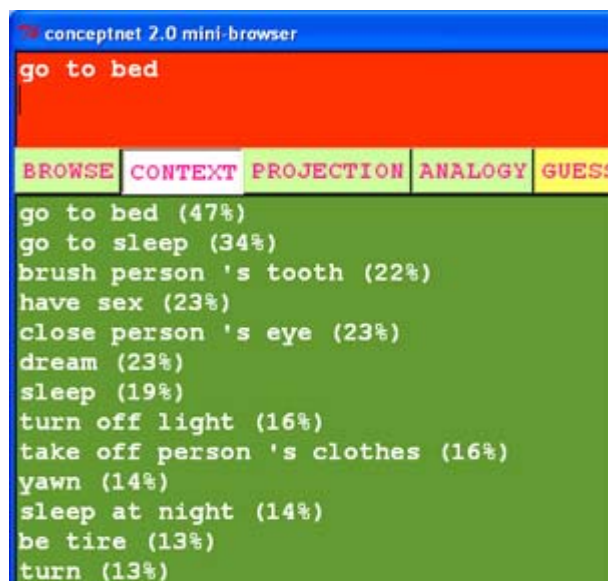


Figura 11 – Exibição dos termos relacionados ao contexto

2. Inferência de encadeamento

A função `FindPathsBetween()` encontra os caminhos entre dois nós. Por exemplo, se digita ‘comprar comida’ e ‘dormir’, a função mostrará todo o caminho percorrido: ‘comprar comida’ → ‘ter comida’ → ‘comer comida’ → ‘sentir-se cheio’ → ‘sentir sono’ → ‘dormir’.

3. Analogia conceitual

A função `GetAnalogousConcepts()` retorna uma lista de conceitos análogos de um conceito. Por exemplo, digitando ‘divã’, a função `GetAnalogousConcepts()` retorna como resultados conceitos com: ‘sofá’, ‘cadeira’, ‘cama’, ‘banco’. A figura 12, uma interface gráfica mostra os resultados obtidos com o uso da função `GetAnalogousConcepts()` da API.

```

conceptnet 2.0 mini-browser
couch
BROWSE CONTEXT PROJECTION ANALOGY GUESS CONCEPT GUESS TOPIC
[~sofa] (41.6629595589)
==LocationOf==> in house (2.37744375108)
==IsA==> piece of furniture (2.37744375108)
==LocationOf==> at home (2.37744375108)
==IsA==> seat (2.37744375108)
==PropertyOf==> comfortable (2.37744375108)
==UsedFor==> sit (2.37744375108)
==LocationOf==> in livingroom (2.37744375108)
==UsedFor==> watch (2.37744375108)
==CapableOfReceivingAction==> find (2.37744375108)
==UsedFor==> relax (2.37744375108)
==CapableOfReceivingAction==> upholster (2.37744375108)
==UsedFor==> comfort (2.37744375108)
==UsedFor==> sit on it (2.37744375108)
==IsA==> comfortable seat (2.37744375108)
==CapableOfReceivingAction==> call (2.37744375108)

```

Figura 12 – Analogias

Essa API possui a responsabilidade no processo de tentar dar a capacidade de raciocínio aos computadores. Ela deve superar as dificuldades de tratar da complexidade da língua natural, pois esta apresenta numerosos fenômenos de linguagem, tais como os advérbios, nomes compostos, discurso direto e indireto, eclipses, etc. além das figuras de linguagem. Também, as múltiplas possíveis interpretações que a língua tem. As palavras são ambíguas e as sentenças permitem múltiplas interpretações (MCCARTHY, 2002). Desse modo, a API trabalha para enviar à aplicação computacional todas as possíveis interpretações e não tem como objetivo apresentar uma única solução certa.

2.4. Considerações finais

Neste capítulo foram abordadas questões relativas ao conceito de senso comum e foi apresentado o projeto Open Mind Common Sense, que está tentando fornecer esse tipo de conhecimento aos computadores.

Foi discutido que, por não terem senso comum, apesar das máquinas poderem fazer atividades especializadas bastantes complexas, como jogar xadrez, elas

ainda não têm a capacidade de compreender e raciocinar sobre fatos bem simples que até uma criança tem.

A definição dada por Marvin Minsky (MINSKY, 1986) ao termo ‘senso comum’ e o utilizado no contexto OMCS também foram assuntos deste capítulo.

Pode-se perceber que o uso da língua natural e a modo colaborativo de obter fatos de senso comum foi um avanço e uma forma de contornar as dificuldades que foram encontradas no projeto Cyc.

Também foi mostrada como é a rede semântica utilizada para fornecer senso comum, a Conceptnet, como gerá-la e como utilizá-la em aplicações computacionais com o auxílio do analisador de língua inglesa, o MontyLingua.

Uma dúvida que pode surgir é saber se os dados inseridos por diversas colaboradores da Web geram uma rede semântica confiável capaz de representar todo o conhecimento de senso comum que um humano possui. O que é e o que não é senso comum? Quanto conhecimento é necessário para conseguir uma base completa de senso comum? Apesar de muitos dados que parecem ser ‘lixo’, acredita-se que elas expressam a opinião e a crença das pessoas. Quanto ao tamanho, o objetivo é um dia chegar a ter os milhões de fatos necessários, mas enquanto isso não é concretizado, dependendo da aplicação, uma máquina com um pouco de senso comum é melhor que uma máquina sem nenhum.

A equipe americana do Projeto OMCS tem como um dos grandes interesses, utilizar fatos de senso comum para facilitar a comunicação multi-língua. Assim surgiu o desafio de criar bases de conhecimento em outros países, sendo o Brasil um deles. Esse assunto será mais detalhado no capítulo seguinte.

CAPÍTULO 3 - O PROJETO OPEN MIND COMMON SENSE NO BRASIL

3.1. Considerações iniciais

Segundo Lieberman (LIEBERMAN E SCHMANDT, 2005), melhorando a habilidade dos turistas, executivos internacionais e pessoas responsáveis pelas relações exteriores a comunicarem efetivamente com pessoas de outras línguas e culturas poderia ocasionar um grande impacto econômico e também contribuir para o entendimento e paz mundial. Com esse discurso, a equipe do projeto Open Mind Common Sense (OMCS) tem como uma das áreas de interesse projetos para facilitar a comunicação multi-língua. Para isso, acredita-se que uma tradução não pode ser feita apenas substituindo as palavras por outras de uma outra língua como nos algoritmos de tradução convencional. Um tradutor deve entender também a semântica envolvida. Para resolver esse problema, pretende-se utilizar o uso do senso comum para ajudar na comunicação entre pessoas que falam idiomas diferentes (LIEBERMAN E SCHMANDT, 2005).

A partir desse propósito surgiu o interesse da equipe do projeto OMCS em criar bases de conhecimento em outros países. Em busca desse objetivo, foi criada a parceria com a atual equipe brasileira do projeto.

Neste capítulo será apresentado o projeto OMCS brasileiro, desenvolvido neste trabalho. Na seção 3.2 será mostrada uma visão geral do projeto e de sua arquitetura; na seção 3.3 será apresentado o site do projeto; na seção 3.4 será mostrada a versão brasileira da rede semântica do projeto OMCS, a ConceptNet brasileira; na seção 3.5 será mostrado o Curupira, o *parser* utilizado para analisar textos em português, substituído o MontyLingua; na seção 3.6 será mostrado como se gera a ConceptNet na

versão brasileira do trabalho; e por fim, na seção 3.7 serão apresentadas as considerações finais desse capítulo.

3.2. Arquitetura do projeto Open Mind Common Sense do Brasil

A figura 13 apresenta esquematicamente a arquitetura para dar uma visão do projeto OMCS no Brasil. Cada um dos elementos será explicado com mais detalhes nas seções posteriores.

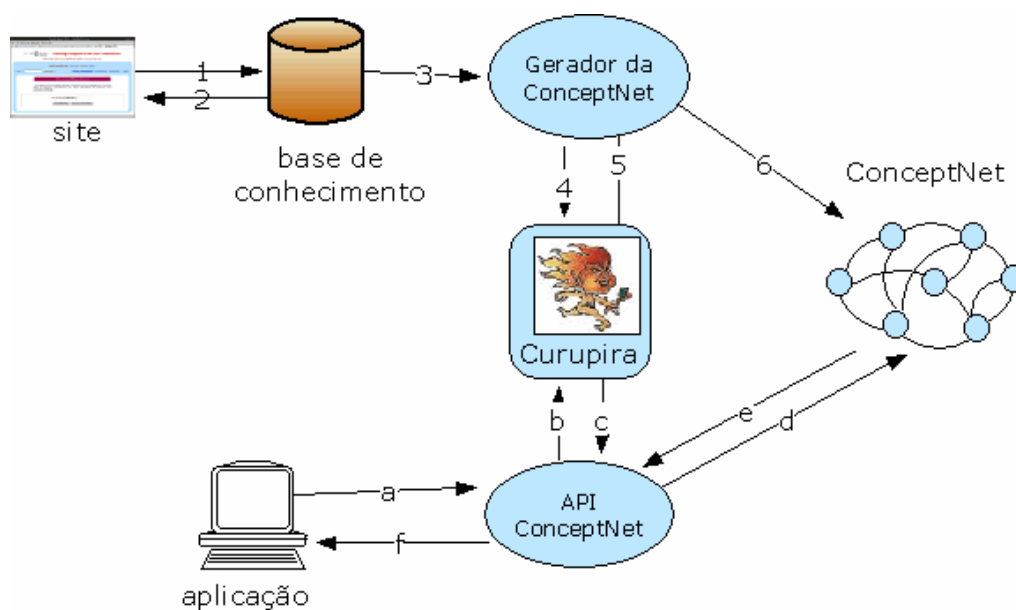


Figura 13 – Arquitetura do projeto OMCS no Brasil

Site: Com o interesse da equipe do projeto OMCS em criar bases de conhecimento em outros países, sendo o Brasil um deles, neste trabalho foi feita a tradução do site para a versão brasileira (<http://lia.dc.ufscar.br/sensocomum>) e em agosto de 2005 ele foi disponibilizado.

O site também é utilizado para o mesmo propósito: colher fatos de senso comum. Atualmente ele apresenta apenas sentenças em forma de *templates*¹¹ (figura 14), não possuindo ainda a forma livre¹². Os dados inseridos são armazenados num

¹¹ Template: Sentenças com formato semi-estruturado.

¹² Forma livre: O texto a ser digitado não possui uma estrutura previamente definida.

banco de dados (seta 1 da figura 15). Esses dados, como na versão original, também são utilizados pelo sistema para formar novas sentenças (seta 2 da figura 15). O funcionamento do site será descrito com mais detalhes na próxima seção, a 3.3.

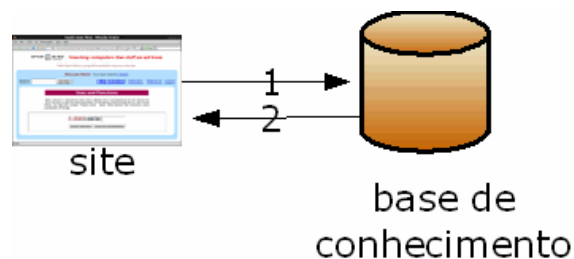


Figura 14 – Aquisição e armazenamento de fatos de senso comum em português

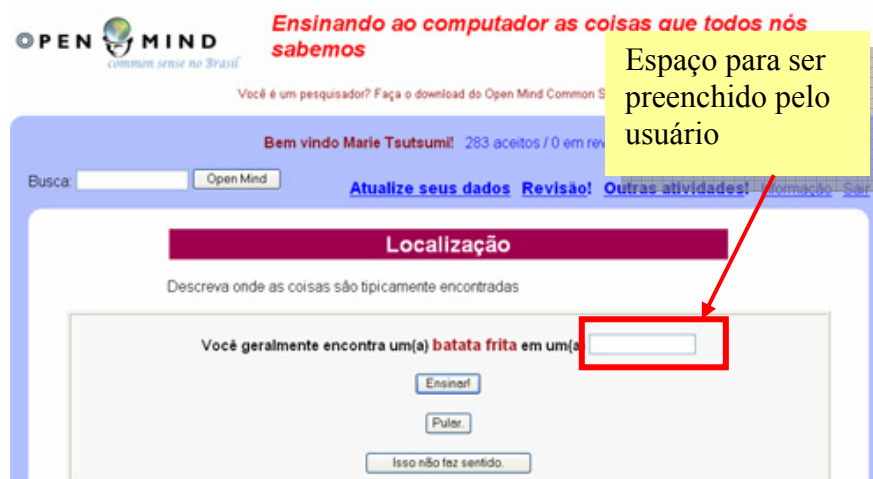


Figura 15 – Sentenças em forma de *templates*

Gerador da ConceptNet: Foi preciso encontrar um analisador de língua portuguesa do Brasil para substituir o analisador de língua inglesa, o MontyLingua. A solução encontrada foi utilizar o programa Curupira, analisador automático de sentenças da língua portuguesa desenvolvido pelo NILC (<http://www.nilc.icmc.usp.br/nilc/index.html>).

A geração da ConceptNet brasileira converte as sentenças armazenadas na base de conhecimento em nós da ConceptNet. Exemplo de uma sentença transformada em um nó:

“Pessoas *comem* quando elas *estão com fome*” → (MotivatioOf, comer, estar com fome)

Esse processo ocorre da seguinte maneira:

1. O gerador da ConceptNet utiliza uma sentença recuperada da base de conhecimento, por exemplo “Pessoas comem quando elas estão com fome” (seta I da figura 16).

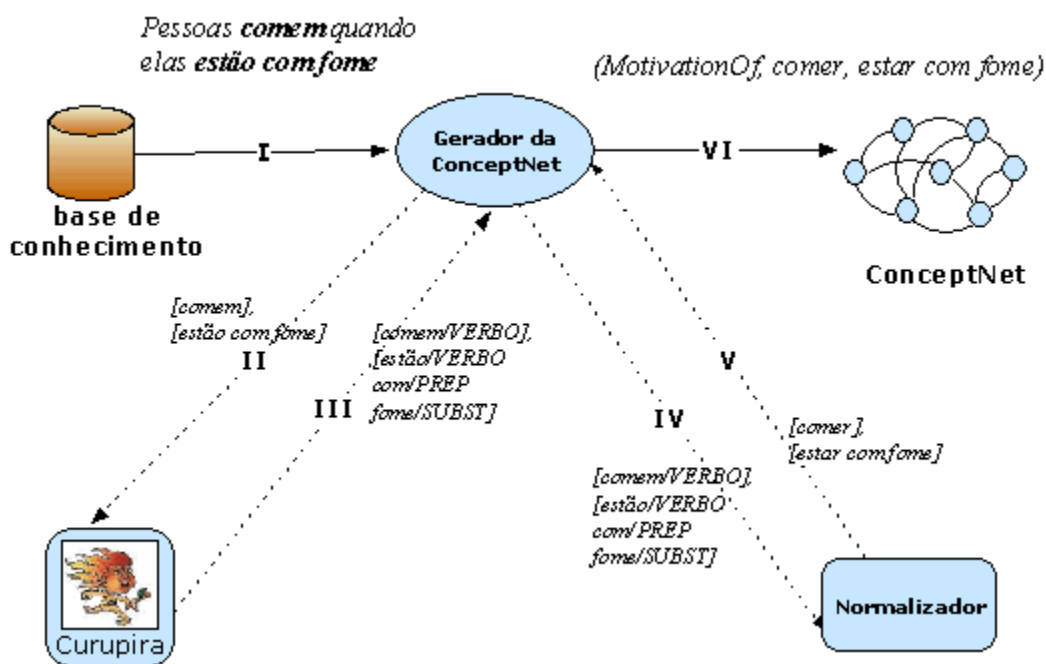


Figura 16 – Gerador da ConceptNet brasileira

2. O gerador então envia os termos da sentença preenchidos pelo usuário para um analisador de língua portuguesa, o Curupira (MARTINS, HASEGAWA, NUNES, 2002) (como se pode ver na seta II da figura 16).

3. Este por sua vez, faz a análise e retorna o resultado para o gerador (seta III da figura 16).

4. Depois o gerador envia as palavras analisadas ao normalizador (seta IV da figura 16), que faz normalização, isto é, cada palavra é convertida em sua forma canônica, assim “comeram” deve se transformar em “comer” e “estão com fome” em “estar com fome”. Como o Curupira não fornece essa opção, um normalizador teve que

ser desenvolvido. Então o normalizador envia o resultado da normalização ao gerador (seta V da figura 16).

5. O gerador então cria mais nó da rede semântica (seta VI da figura 16).

O funcionamento do gerador da ConceptNet brasileira será descrito com mais detalhes na seção 3.4.

API brasileira: Neste trabalho também foi feita a modificação da API (*Application Programming Interface*), pois ela passou a chamar os métodos do Curupira. O funcionamento da API, do mesmo modo que a versão em inglês, é descrito em 5 passos:

1. Inicialmente o usuário da aplicação computacional digita um texto, por exemplo, “Jantaremos em casa?” (seta “a” da figura 17).

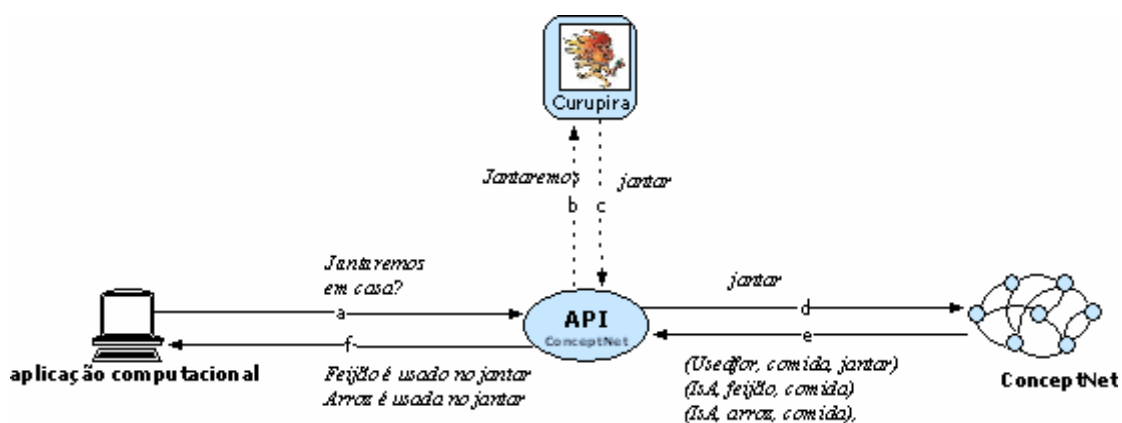


Figura 17 – Funcionamento da API brasileira

2. Então, a API chama métodos do Curupira, ao invés do MontyLingua como no projeto original. O Curupira então faz a análise do texto digitado (seta “b” da figura 17), e retorna o texto um formato adequado (seta “c” da figura 17) para que a API possa utilizar. Por exemplo, depois da análise da palavra “Jantaremos”, o Curupira retorna a forma normal de “Jantaremos” que é “jantar” e envia de volta para a API.

3. A API utiliza esse termo, agora na forma canônica, como a expressão de busca na rede semântica (seta “d” da figura 17).

4. Depois, a API obtém os nós relacionados ao termo considerando o contexto, (seta “e” da figura 17).

5. Então a API envia os nós em um formato legível ao usuário (seta “f” da figura 17). Depois, a aplicação computacional mostra ao usuário, textos em português com algumas sugestões relacionadas ao texto digitado pelo usuário inicialmente.

O funcionamento do gerador da ConceptNet brasileira será descrito com mais detalhes na seção 3.5.

Com esse sistema apresentado, têm sido desenvolvidos estudos relacionados à possibilidade de utilização deste estudo para apoiar as práticas de aprendizagem eletrônica, pois se acredita que considerando os fatos de senso comum dos alunos é possível auxiliar instrutores na preparação de ações de aprendizagem para produzir cursos mais adequados e conseqüentemente, com maior aceitação (ANACLETO ET AL., 2006c).

Também têm sido investigados os efeitos que as cores causam no comportamento das pessoas. Sabendo o que a maioria das pessoas sentem quando estão diante de uma determinada cor, acredita-se que é possível melhorar também a impressão que elas têm diante de um sistema Web, dependendo das cores utilizadas (BUCHDID, SILVEIRA, SILVA, 2005; SILVEIRA, BUCHDID, SILVA, 2005).

Além dessas pesquisas citadas acima, têm sido feito estudos para identificar aspectos culturais expressos nos fatos de senso comum armazenados na base de dados (ANACLETO ET AL., 2006A). Acredita-se entendendo as diferenças culturais, é possível auxiliar a projetar interfaces mais adequadas às pessoas pertencentes a uma determinada

cultura; e também oferecer soluções computacionais para melhorar a comunicação entre pessoas de culturas diferentes. Esta última linha de pesquisa citada será o assunto abordado neste trabalho.

3.3. O Site Open Mind Common Sense no Brasil

Acreditando que a colaboração do público geral poderia ajudar a criar a base de conhecimento e inserir um conjunto expressivo de conhecimento de senso comum parece não ser uma tarefa muito simples de se fazer manualmente, a equipe brasileira do projeto OMCS também adotou a mesma abordagem utilizada pela equipe americana: obter fatos de senso comum através da colaboração através da Web.

Para isso, o site, codificado em JSP (*Java Server Pages*), foi traduzido para o português neste trabalho, e em agosto de 2005 foi lançado o OMCS Brasil (<http://lia.dc.ufscar.br/omcs>), a primeira versão do site em uma língua não inglesa. Nesta versão, os fatos continuam sendo escritos em língua natural, e como o objetivo é colher fatos de senso comum do público brasileiro, as sentenças a serem preenchidas agora são apresentadas em português do Brasil. Um banco de dados MySQL é utilizado para armazenar as sentenças e o relacionamento existente entre os fatos inseridos.

Do mesmo modo que o site original, para que as pessoas comecem a contribuir, é preciso entrar no site e preencher um pequeno cadastro, fornecendo além dos dados pessoais do site original, *login* e senha, alguns dados adicionais para estudar o perfil dos colaboradores foi solicitado o site brasileiro, como o grau de escolaridade, sexo, idade, estado que reside, como mostra a tabela 3 a seguir.

Tabela 3 – Mais detalhes dos perfis dos usuários no site brasileiro

Site original	Site brasileiro
Nome	Nome
E-mail	E-mail
<i>Login</i>	<i>Login</i>

Senha	Senha
-	Idade
-	Sexo
-	Escolaridade
-	País
-	Estado
-	Cidade
CEP	-
Áreas de interesse	-

Feito o cadastro, é só completar as sentenças em português apresentadas aleatoriamente, como por exemplo, “Uma coisa que você pode encontrar em um(a) quadra é um(a) ___”; “Um outro jeito de dizer ter calma, é ___”. O site também permite ao usuário escolher uma atividade específica para inserir dados. Atualmente, foram escolhidas 8 atividades das 29 do site original, consideradas mais interessantes, além da adição de outras 2 atividades (*Cores* subdividida em 3 e *Saúde* subdividida em 14), voltadas exclusivamente aos assuntos pertinentes às pesquisas que seriam realizadas pela da equipe OMCS no Brasil, totalizando 10 atividades, como mostrado na tabela 8.

Além da lista com os nomes dos usuários que mais colaboraram para promover uma disputa entre eles, foram feitas divulgações em mídias para atrair usuários novos e envolvê-los com o projeto. Foram exibidas reportagens em jornais de circulação nacional (O Estado de São Paulo (SILVA, 2005A), Jornal da Tarde (SILVA, 2005B), e Correio Brasiliense (CERATTI, 2005)), sites de notícias e divulgação científica e tecnológica (IDG Now (IDG NOW!, 2005), IG - Último Segundo (ÚLTIMO SEGUNDO, 2005), Revista Oi (LACERDA, 2005), Comciência (COMCIÊNCIA, 2005), UFRGS Clipping de Jornal (CLIPPING DE JORNAL, 2005)), e uma reportagem em um programa de televisão regional (Eptv Notícias - Rede Globo). Como resultado, foram obtidos por volta de 100 mil fatos coletados de mais de 1000 usuários¹³ desde o lançamento do site (7 meses) de pessoas de diversas faixas etárias e dos mais variados locais. O perfil dos

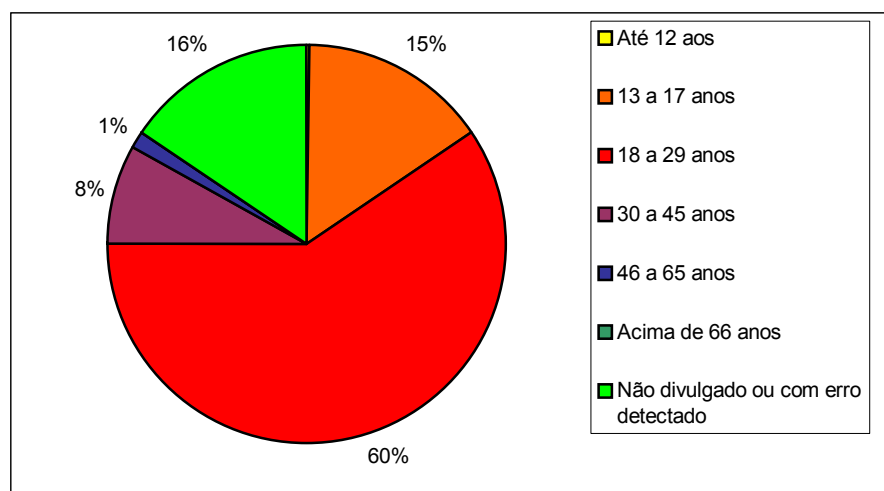
¹³ Dados obtidos em 12 de março de 2006

usuários que se cadastram o site é mostrado nas tabelas 4, 5, 6 e 7 e nos gráficos 1, 2, 3 e 4¹⁴.

Tabela 4 – Faixa etária dos colaboradores

Faixa etária	Número de colaboradores
Até 12 aos	3
13 a 17 anos	163
18 a 29 anos	631
30 a 45 anos	86
46 a 65 anos	13
Acima de 66 anos	1
Não divulgado ou com erro detectado ¹⁵	165
Total	1062

Gráfico 1 – Porcentagem de cada faixa etária dos colaboradores



O que se pode notar é que a maior número de colaboradores é jovem. A maioria está na faixa entre 18 a 29 anos, representando quase 60% dos cadastrados e os adolescentes na faixa etária de 13 a 17 anos aparecem em segundo lugar, representando mais de 15% dos cadastrados.

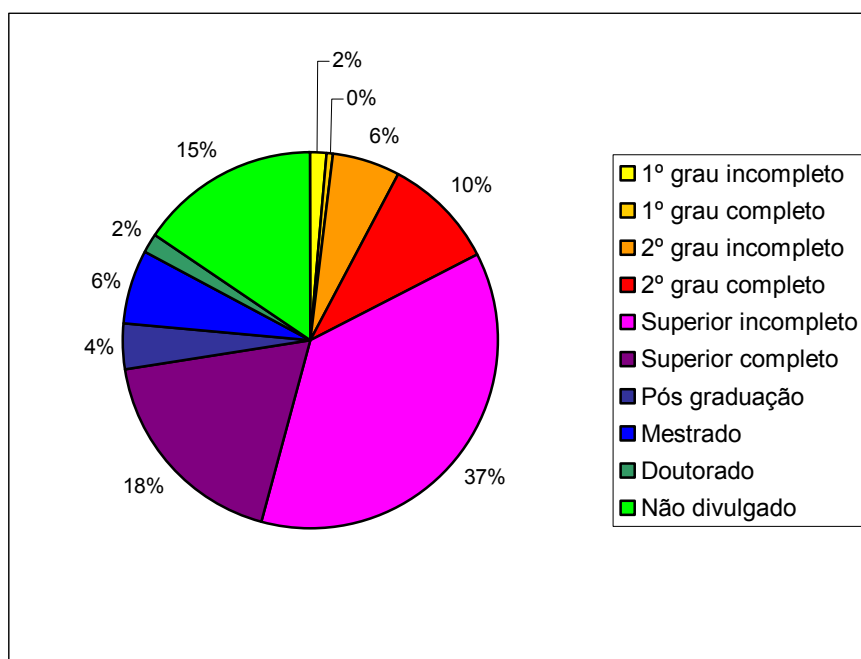
¹⁴ Dados obtidos em 12 de março de 2006

¹⁵ Erro detectado: por exemplo, faixa de 13 a 17 anos cursando doutorado

Tabela 5 – Escolaridade dos colaboradores

Escolaridade	Número de colaboradores
1º grau incompleto	16
1º grau completo	4
2º grau incompleto ¹⁶	63
2º grau completo	103
Superior incompleto ¹⁷	389
Superior completo	194
Pós graduação	42
Mestrado	68
Doutorado	19
Não divulgado	164
Total	1062

Gráfico 2 – Porcentagem dos níveis de escolaridade dos colaboradores



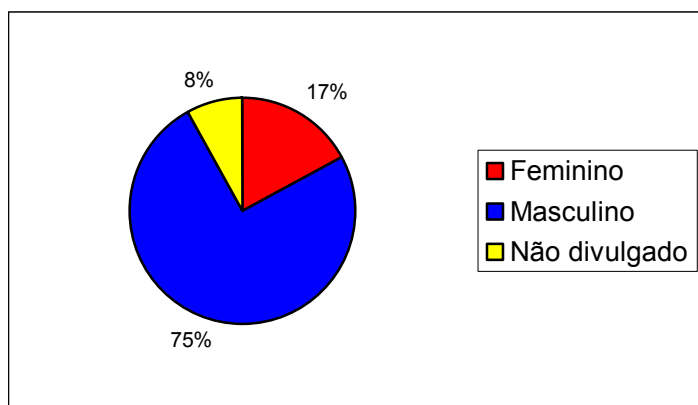
O que se pode perceber é que os estudantes de graduação apresentam o maior número de usuários, representando mais de 36% dos cadastrados.

¹⁶ Quando não foi definido se a escolaridade é completo ou incompleto, foi considerada incompleta.

Tabela 6 – Gênero dos colaboradores

Gênero	Número de colaboradores
Feminino	184
Masculino	791
Não divulgado	87
Total	1062

Gráfico 3 – Porcentagem dos gêneros dos colaboradores



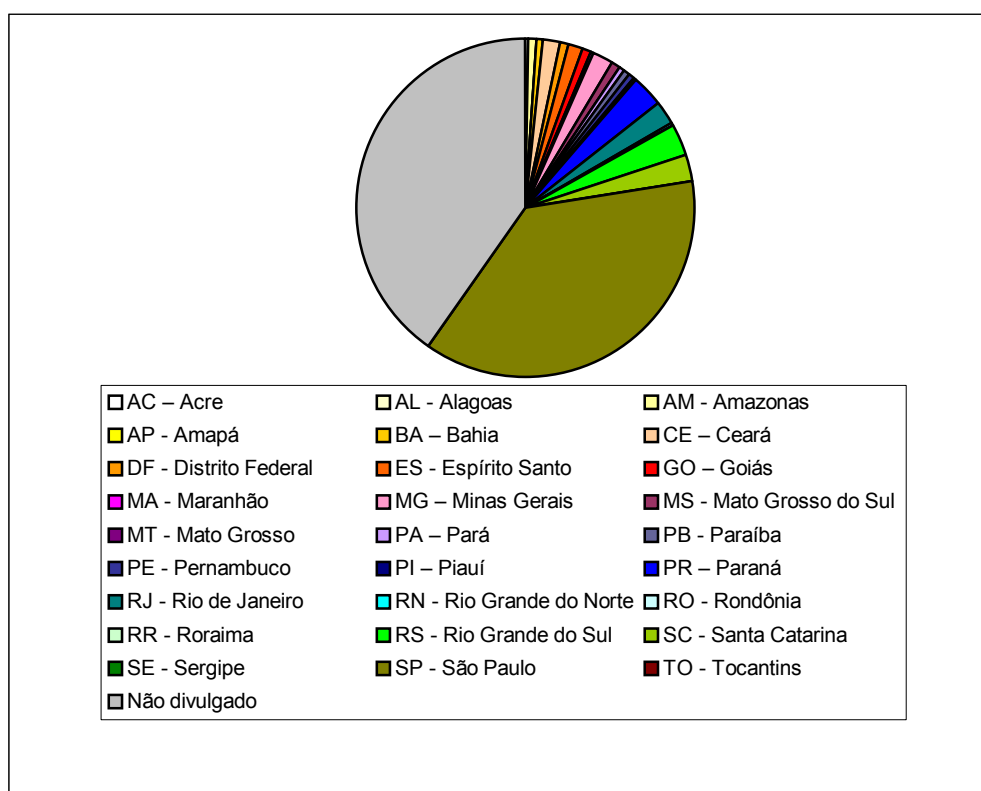
A maioria absoluta dos interessados em colaborar com o projeto é formada por homens, que representam mais de 74% dos cadastrados.

Tabela 7 – Estado de origem dos colaboradores

Estado	Número de colaboradores
AC – Acre	3
AL - Alagoas	0
AM - Amazonas	9
AP - Amapá	0
BA – Bahia	7
CE – Ceará	15
DF - Distrito Federal	11
ES - Espírito Santo	13
GO – Goiás	11
MA - Maranhão	1
MG – Minas Gerais	22
MS - Mato Grosso do Sul	7
MT - Mato Grosso	2
PA – Pará	4
PB - Paraíba	6
PE - Pernambuco	7
PI – Piauí	2

PR – Paraná	34
RJ - Rio de Janeiro	24
RN - Rio Grande do Norte	1
RO - Rondônia	0
RR - Roraima	0
RS - Rio Grande do Sul	32
SC - Santa Catarina	27
SE - Sergipe	0
SP - São Paulo	395
TO - Tocantins	1
Não divulgado	428
Total	1062

Gráfico 4 – Ilustração da porcentagem dos estados de origem dos colaboradores



Infelizmente o grupo que mais aparece no banco de dados dos cadastrados é de usuários que não divulgaram o estado de origem. Entre os que divulgaram, a maior número de usuários é do estado de São Paulo, representado mais de 37% dos cadastrados. Também pode se notar que a região sul e sudeste são os estados que apresentam mais cadastrados que os demais estados.

No site brasileiro desenvolvido neste trabalho, todos os fatos inseridos pelos usuários passam por uma revisão manual. Deste modo elimina-se os palavrões e palavras com erros ortográficos. Os dados aceitos são re-utilizados pelo sistema para compor novas sentenças, estas serão utilizadas para obter outros novos dados. Para um melhor entendimento, um exemplo é apresentado na figura 18, onde é mostrado que o termo “batata frita” preenchido pelo usuário é utilizado para formar novas sentenças, como por exemplo, “Você encontra uma batata fria em um ___”.

Para que possa haver a retro-alimentação, os dados inseridos pelos usuários devem estar de acordo com as regras apresentadas na tabela 8. Os campos dos templates sinalizados por (___) representam campos que são preenchidos automaticamente pelo site através da retro-alimentação. Já os campos sinalizados por [___] representam campos que serão preenchidos pelo colaborador do site e as abreviações que são encontradas dentro deles correspondem aos tipos de entradas que é esperado. Os significados dessas abreviações são mostrados na tabela 9.

Tabela 8 – Templates das atividades do site Open Mind Common Sense no Brasil

ATIVIDADE	TEMPLATE
Coisas	Uma coisa que você pode encontrar em um(a) (L) é um(a) [S]
Sentenças	Você geralmente quer um(a) (S) para [VI]
Usos	Um(a) (S) é usado para [VI]
Localização	Você geralmente encontra um(a) (S) em um(a) [L]
	Você geralmente encontra um(a) [S] em um (a) (L)
Pessoas	Pessoas [VP] quando elas [VP]
Analogia	Um outro jeito de dizer (VI) é [VI]
Problemas	Quando se tenta (VI) um problema encontrado pode ser [VI]
Ajuda	Quando pessoas (VP) uma forma de ajudar é [VI]
Cores	A cor (cor gerada pelo sistema) faz ter vontade de [VI]
	A cor (cor gerada pelo sistema) me lembra um [VI]
	(S) me faz lembrar a cor [S]
Saúde	Para cuidar de alguém doente em casa deve-se [VI]
	Antes de começar a cuidar de alguém doente em casa deve-se [VI]
	Para poder cuidar de alguém doente em casa é preciso ter [S]
	Para poder cuidar adequadamente de alguém doente em casa é preciso estar [E]
	Para dar o remédio a alguém doente deve-se [VI]
	O que leva alguém cuidar de um doente em casa é [S]
	Uma pessoa doente gosta de [VI]
	Para conseguir estar (E) deve-se [VI]
	As pessoas cuidam de um doente em casa para [VI]
	(S) ajuda no cuidado da pessoa doente porque [J]

[S] pode ser um dos sintomas de [S]
Em casa, quando a pessoa tem (S) uma forma de ajudá-la é [VI]
Em casa, quando a pessoa tem (S) um tratamento alternativo ao remédio pode ser [VI]
É (difícil fácil) cuidar de alguém doente em casa porque [J]

The figure consists of two screenshots of the Open Mind website interface, illustrating a feedback loop. Both screenshots show the user profile 'Marie Tsutsumi' with 283 accepted items and 0 in review.

Top Screenshot: 'Coisas' section
 The user is prompted to 'Conte-nos as coisas que você encontra em seu dia-a-dia'. The question is: 'Uma coisa que você pode encontrar em um(a) restaurante popular é um(a)'. The user has entered 'batata frita' in the input field. Below the input are buttons for 'Ensinar!', 'Pular.', and 'Isso não faz sentido.'

Bottom Screenshot: 'Localização' section
 The user is prompted to 'Descreva onde as coisas são tipicamente encontradas'. The question is: 'Você geralmente encontra um(a)'. The user has entered 'batata frita' in the input field. Below the input are buttons for 'Ensinar!', 'Pular.', and 'Isso não faz sentido.'

A red arrow points from the 'batata frita' input in the top screenshot to the 'batata frita' input in the bottom screenshot, indicating that the system has used the user's input from the first step to inform the second step.

Figura 18 – Exemplo de retro-alimentação

Tabela 9 – Abreviações utilizadas nos *templates*

Letra	Significado
S	substantivo no singular + complementos. Ex.: carro, casa velha, etc
E	estado físico ou emocional. Ex.: saudável, preparado, etc.
L	substantivo que denote um local. Ex.: praça, sala de jantar, etc
VI	verbo no infinitivo + complementos. Ex.: nadar, fazer compras, etc.
VP	verbo conjugado na 3ª pessoa do plural + complementos. Ex.: andam rápido, etc.
J	justificativa composta de uma sentença em língua portuguesa.

O esquema foi montado de acordo com que se julgou ser a forma mais comum do público completar um determinado *template*. Por exemplo, para exibir ao usuário o *template* “Uma coisa que você pode encontrar em um(a) (L) é um(a) [S]”, será recuperado da base aleatoriamente um lugar(L), por exemplo, a palavra “casa”, formando assim a sentença “Uma coisa que você pode encontrar em um(a) **casa** é um(a) ____”. Espera-se que o usuário então preencha com algum substantivo [S].

Para um melhor entendimento o esquema é apresentado na figura 19 em forma de grafos. O sentido das setas indica que dados fornecidos na atividade de origem são utilizados na formação da nova sentença da atividade de destino. Por exemplo, os dados fornecidos na atividade “Coisas” são utilizados para compor as frases utilizadas nas frases das atividades “Sentenças”, “Usos” e “Localização”. Assim, se na atividade “Coisas” um usuário entra com o fato “Uma coisa que você geralmente encontra em uma casa é um(a) cortina”, como a palavra “cortina” é um substantivo e portanto obedece a regra esperada “Uma coisa que você pode encontrar em um(a) (L) é um(a) [S]”, essa palavra será utilizada para gerar as sentenças de outras atividades como:

Você geralmente quer uma **cortina** para _____ (atividade “Sentenças”)

Um(a) cortina é usado(a) para _____ (atividade “Usos”)

Você geralmente encontra um(a) **cortina** em um(a) _____ (atividade Localização)

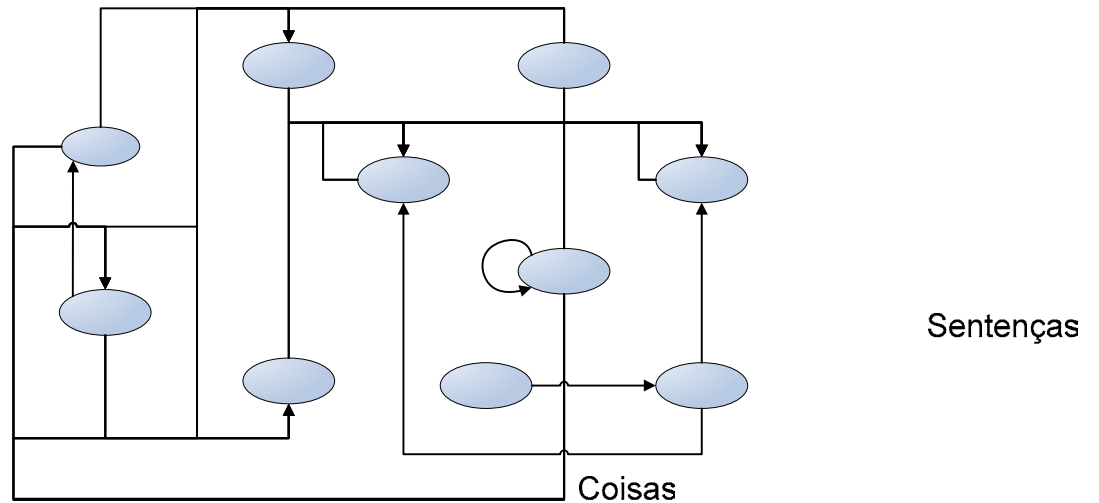


Figura 19 – Esquema de retro-alimentação da base de dados OMCS no Brasil

Como nem sempre o que os usuários acham natural é o que se espera, muitas vezes os *templates* podem ser preenchidos de forma diferente das regras de retro-alimentação estabelecidas, mas que fazem sentido a frase. Então o fato é aceito, porém não é utilizado na retro-alimentação.

A figura 20 mostra uma tela que pode ser visualizada pelo revisor, onde se encontra a opção de aceitar ou rejeitar os fatos e uma opção para utilizar os fatos para retro-alimentação.

Opção para aceitar ou rejeitar os fatos			Revisar contribuição	Opção para retro-alimentação	
Acceptar	Manter	Rejeita	Conhecimento	Retro-Alimentar	Data
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	lembra um(a) banana	<input checked="" type="checkbox"/>	2006-03-12 13:07:46.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	jaula me faz lembrar a cor cinza	<input checked="" type="checkbox"/>	2006-03-12 13:07:26.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	faz ter vontade de beber suco de uva	<input checked="" type="checkbox"/>	2006-03-12 13:07:19.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	lembra um(a) calcinha	<input checked="" type="checkbox"/>	2006-03-12 13:07:09.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	faz ter vontade de bater em alguém	<input checked="" type="checkbox"/>	2006-03-12 13:06:55.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	animal silvestre me faz lembrar a cor amarela	<input checked="" type="checkbox"/>	2006-03-12 13:06:30.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	lembra um(a) árvore	<input checked="" type="checkbox"/>	2006-03-12 13:06:41.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	lembra um(a) machucado	<input checked="" type="checkbox"/>	2006-03-12 13:02:23.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	faz ter vontade de beber água	<input checked="" type="checkbox"/>	2006-03-12 12:53:05.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	todinho me faz lembrar a cor marrom	<input checked="" type="checkbox"/>	2006-03-12 12:53:22.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	souden me faz lembrar a cor rosa	<input checked="" type="checkbox"/>	2006-03-12 12:53:27.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	faz ter vontade de beber água	<input checked="" type="checkbox"/>	2006-03-12 12:53:38.0
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	motorista de onibus me faz lembrar a cor cinza	<input checked="" type="checkbox"/>	2006-03-12 12:53:47.0

Figura 20 – Tela com opção de retro-alimentação

Esse mecanismo que permite escolher se os dados preenchidos em uma sentença serão utilizados na retro-alimentação é encontrado somente no site brasileiro. Esse mecanismo foi desenvolvido devido à ausência de um corretor ortográfico automatizado. Desse modo, evita-se que os dados inseridos incorretamente possam atrapalhar a formação de uma nova sentença.

3.4. A ConceptNet

Da mesma maneira que as sentenças obtidas do site original do OMCS, as sentenças obtidas do site brasileiro também não são diretamente computáveis. Os fatos inseridos em português são também convertidos para aquele formato de declarações – descrito no capítulo anterior. O conjunto de declarações forma a rede semântica, a ConceptNet (LIU E SINGH, 2004A), cujo formato pode ser entendido por um computador. Na figura 21, uma pequena parte da rede semântica em português poder ser observada.

Nesta versão, como na versão original, uma declaração consiste de dois “conceitos” e “relações” estabelecidas entre eles, mais a “frequência” que a declaração aparece no banco de dados e a “inferência” realizada para gerar a declaração, e funcionam da mesma maneira:

- **Conceito:** Um conceito representa os nós na rede semântica. Cada nó agora é um fragmento em português: verbo, frase nominal, frase preposicional e frase adjetiva, como exemplificados na tabela 10.

Tabela 8 – Tipos de conceitos ou nós

Verbo (Evento)	Frase nominal (Coisa)	Frase preposicional (Lugar)	Frase Adjetiva (Propriedade)
Dormir cedo	Mesa da cozinha	Na casa	Barulhento
Acordar de manhã	despertador	Na mesa	Pequeno

- **Relação:** é relação semântica que existe entre dois conceitos. A relação é representada por uma **ligação** entre os nós (figura 21). Estão sendo utilizadas as relações da ConceptNet original (tabela 2 da página 17).

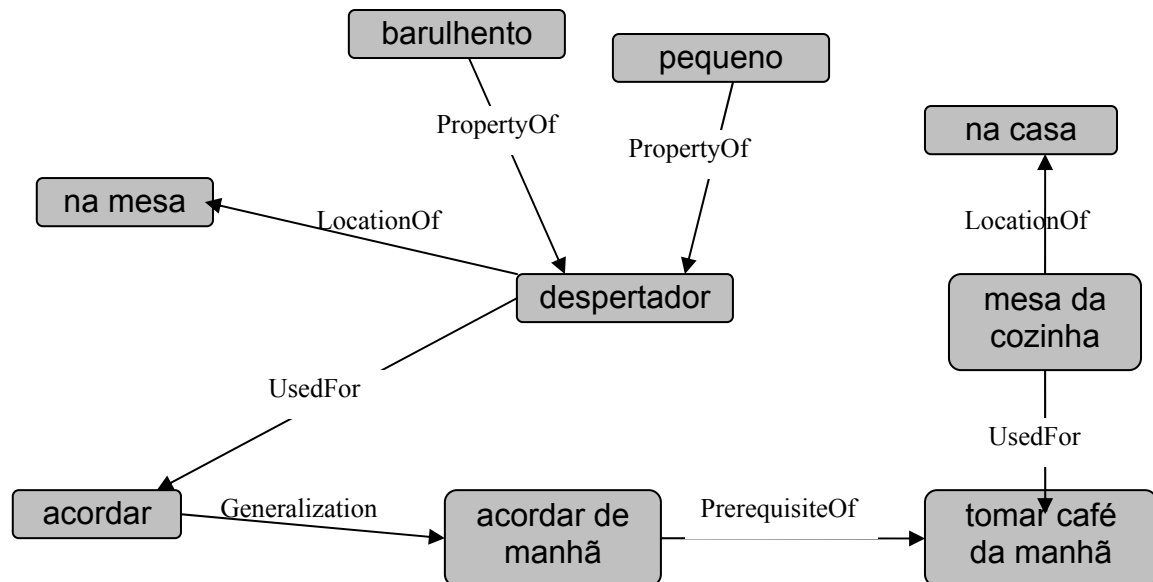


Figura 21 – Trecho da ConceptNet brasileira

- **Frequência (f):** é o número de vezes que a declaração aparece em toda base do OMCS. Por exemplo: Na declaração (**UsedFor** ‘despertador’ ‘acordar’ ‘f=2;i=0’), o valor da frequência é igual a 2 (f=2), isso significa que existem duas sentenças na base OMCS dizendo que o despertador é usado para acordar.

- **Inferência (i):** é o número de declarações que foram necessárias para criar a nova declaração. Por exemplo, da sentença ‘Um pardal é um pássaro’, é extraída a declaração (**IsA**, ‘pardal’, ‘pássaro’ ‘f=1; i=0’). Como esta declaração foi gerada diretamente de uma sentença, não foi necessária nenhuma outra declaração para criá-la, então valor da inferência é zero (i=0). A partir dela, pode ser inferida uma nova declaração como (**CapableOf**, ‘pardal’ ‘voar’ ‘f=1; i=1’). A partir da declaração “Um pardal é um pássaro” foi necessária utilizar uma outra declaração “Um pássaro é capaz

de voar” para gerar a declaração “Um pardal é capaz de voar”, por isso, o valor da inferência é 1 ($i=1$).

Essas declarações são geradas a partir da base do OMCS brasileira com o auxílio de um analisador de língua portuguesa, o Curupira (MARTINS, HASEGAWA, NUNES, 2002), descrito na seção seguinte.

3.5. O Curupira

Como o MontyLingua analisa sentenças em inglês, foi preciso procurar um programa que realizasse essa mesma função, porém, para as sentenças escritas em português. A solução encontrada foi o uso do programa Curupira (MARTINS, HASEGAWA, NUNES, 2002).

Curupira é um *parser*, ou seja, um analisador de sentenças de língua portuguesa desenvolvido pelo NILC (<http://www.nilc.icmc.usp.br>). Trata-se de um analisador sintático da língua portuguesa, que fornece todas as possíveis análises sintáticas para qualquer sentença escrita em português brasileiro.

Em relação à concepção de sentença, cabe dizer que o Curupira interpreta como "sentença" toda e qualquer seqüência de caracteres isolada por dois delimitadores, representados por um conjunto de caracteres especiais de pontuação ou de comando. São eles: o ponto, o ponto-e-vírgula, os dois-pontos, as reticências, o ponto-de-interrogação, o ponto-de-exclamação e o travessão, entre os sinais de pontuação; e os marcadores de início-de-linha, de fim-de-linha, de início-de-parágrafo, de fim-de-parágrafo, de início-de-coluna, de fim-de-coluna, de início-de-página e de fim-de-página, como caracteres de comando.

O tamanho ou a variedade do intervalo entre dois delimitadores não importa ao Curupira, que está preparado para a análise de sentenças de qualquer comprimento.

Não há, portanto, restrições quanto ao número de palavras e/ ou de caracteres da sentença de entrada. Mas é preciso observar que sentenças muito longas degradam o desempenho da ferramenta, que poderá gerar problemas em ambientes computacionais menos robustos, pois ele procura todas as possíveis soluções para cada palavra da sentença.

O Curupira procura etiquetar, sintaticamente, cada um dos itens lexicais da sentença, por menores e menos expressivos que sejam.

Supõe-se que a sentença de entrada esteja correta. Depois o software fornece, não apenas uma, mas todas as possibilidades sintáticas para uma dada sentença de entrada. Para cada entrada, o Curupira retorna:

- a classe das palavras (substantivo, adjetivo, numeral, nome próprio, abreviatura, sigla, prefixo, interjeição, conjunção, preposição, artigo, advérbio, verbo, pronome);
- as subclassificações pertinentes a cada classe (pronome pessoal, possessivo, demonstrativo, indefinido, interrogativo, de tratamento, em relação aos pronomes, por exemplo; ou numeral multiplicativo, cardinal, ordinal e fracionário, em relação aos numerais);
- o gênero, sempre que pertinente (masculino, feminino, ou uniforme);
- o número, sempre que pertinente (singular, plural ou invariável);
- as formas verbais (presente do indicativo, futuro do subjuntivo, etc.);
- o número e a pessoa, para as formas verbais e para os pronomes pessoais (primeira pessoa do singular, segunda pessoa do plural, etc.);
- o grau, sempre que pertinente (positivo, aumentativo, diminutivo);

- a transitividade, para os verbos (transitivo direto, intransitivo, etc.);
- a regência, para os substantivos, adjetivos, verbos e advérbios que exigem complemento preposicionado;
- os advérbios (de tempo, de modo, de intensidade, etc.).

O dicionário (ou léxico) utilizado pelo Curupira trata-se de uma base de dados lexical, não restrita em relação a nenhum domínio, onde estão representadas as palavras simples e compostas da língua portuguesa. Este léxico soma hoje cerca de 1,5 milhão de entradas, em formato de texto. O Curupira também é integrado por um conjunto de lexias complexas (locuções prepositivas, locuções conjuncionais, locuções e expressões latinas, etc.).

Para tanto, a sentença é processada seguindo os seguintes passos:

- Tokenização da sentença - a frase é quebrada em itens lexicais, ou tokens.
Exemplo: a frase “Comi cinco frangos assados”, é separada em “Comi”, “cinco”, “frangos” e “assados”.
- Recuperação das possíveis categorias e atributos dos itens lexicais.
Exemplo: a palavra “assado” pode ser substantivo (Comi uns **assados**), verbo (Os frangos foram **assados** depressa) ou adjetivo (Comi cinco frangos **assados**).
- Desambiguação lexical dos itens lexicais do tipo palavra que contenham mais de uma categoria distinta. Para cada item lexical são aplicadas regras de desambiguação que utilizam, além das categorias do próprio item, informações dos itens lexicais vizinhos à esquerda e à direita. Exemplo: Como a palavra “assados” aparece depois do substantivo “frangos”, entende-se que “assados” seja um adjetivo.

- Análise Sintática dos itens lexicais - o *parser* continua trabalhando enquanto houver sucesso nos casamentos das regras e itens lexicais, retornando zero ou mais resultados.
- Os resultados são formatados e armazenados em um arquivo de saída. A seguir, um trecho do conteúdo gerado no arquivo de saída:

```
0) Escreva --> VERBO[escrever]-> Bitrans Intrans Pron TransDir
TransInd
1) a --> PREPOSICAO
2) sua --> PROPOS
3) frase --> SUBST
4) aqui --> ADV
5) . -->
6) #>>>>>>>>> DELIMITADOR
Número de itens: 6
```

As funções do Curupira encontram-se em um arquivo DLL, portanto, não se pode fazer modificações no código do programa para alterar ou acrescentar funcionalidades. Para a manipulação do programa, os nomes das funções, os tipos das variáveis de entrada e de saídas foram disponibilizados, como listado seguir:

```
BOOL InicializaCurupira()
{
    WORD versao = 0;
    CString strDic = m_strDir + m_strDic; // caminho e nome do dicionário
    utilizado pela dll.
    CString strDll = m_strDir + "minhasdlls.dll"; // nesta versão não está
    sendo utilizada.
    CString strSin = m_strDir + "saida.sin"; // nome do arquivo de saída com
    os resultados.
    RevVersao(&versao); // retorna a versão da dll.
    if (versao != 0x0201) // verifica se a versão está correta.
        return FALSE;
    if (RevInic(strDll, strSin, szSemelhantes)) // Inicializa as estruturas
    internas da dll.
        return FALSE;
    if (RevMensagens()) // Monta as mensagens utilizadas pelo revisor
    gramatical.
        return FALSE;
    if (RevAbrirDicBas(strDic, 1)) // Carrega o dicionário básico.
        return FALSE;
    return TRUE;
}
```

}

Para que essas funções do Curupira pudessem ser acessadas pelo Gerador e pela API da ConceptNet, escritos na linguagem Python, uma nova classe teve que ser implementada para tornar possível essa comunicação, que serve de interface entre o Curupira e as funções do Gerador e da API da ConceptNet. A figura 22 mostra como ocorre essa comunicação:

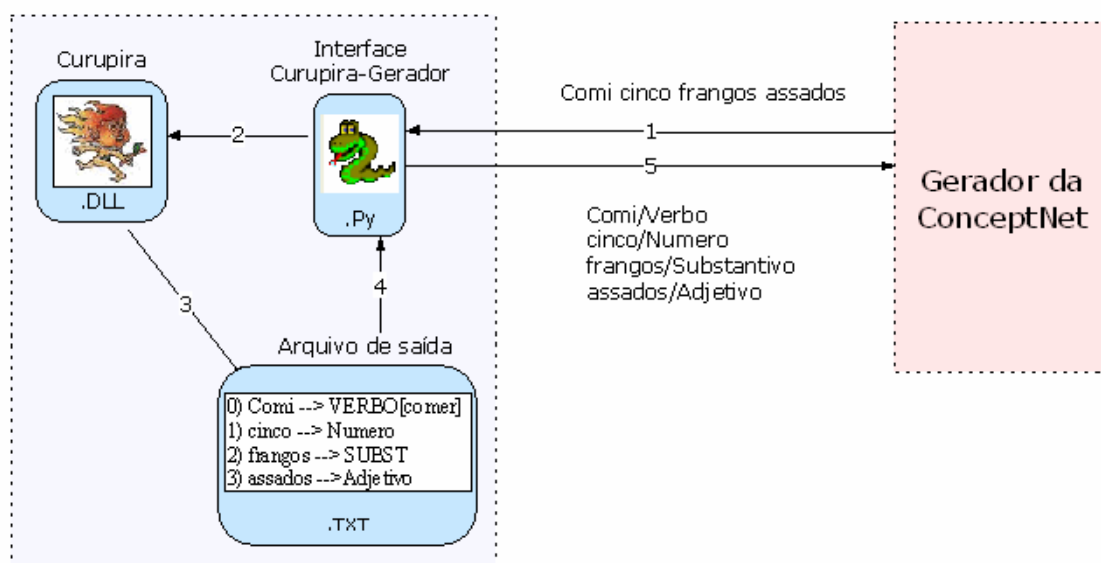


Figura 22 – Comunicação entre o Curupira e as funções de manipulação da ConceptNet

O Gerador ou a API da ConceptNet envia uma frase à nova classe (seta 1), por exemplo, “Comi cinco frangos assados”.

A nova classe acessa as funções do Curupira enviando a frase como parâmetro (seta 2).

O Curupira gera um arquivo de saída em formato texto com a frase analisada (seta 3).

Os dados gerados no arquivo são lidos pela nova classe (seta 4), que envia para o Gerador ou API (seta 5).

E por fim, as funções do Gerador ou da API utilizam o texto analisado no formato que essas funções utilizam.

Como o Curupira não apresenta a forma canônica, um normalizador teve que ser construído para realizar tal tarefa. Para isso, foi utilizado o dicionário DELAF (<http://www.nilc.icmc.usp.br:8180/unitex-pb>), que pode ser encontrado no Anexo B.

O Normalizador foi escrito em Python, e tem como objetivo utilizar o texto analisado pelo Curupira e transformar as palavras na forma normal, como apresentado na figura 23.

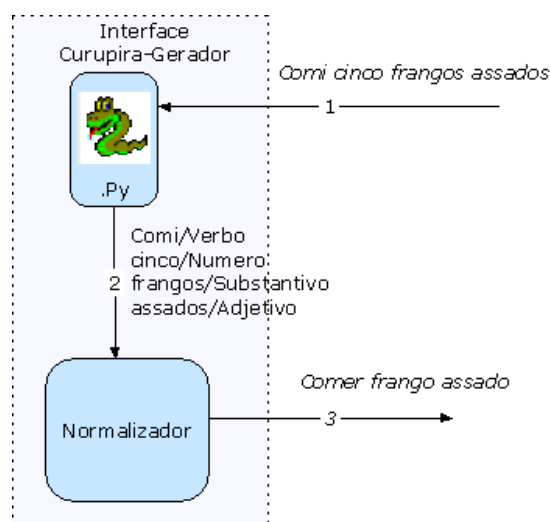


Figura 23 – Normalizador

O normalizador procura no dicionário, a forma normal para uma determinada palavra de acordo com a classificação feita pelo Curupira. Por exemplo, na frase “Comi cinco frangos assados”, o Curupira classificaria a palavra “assados” como um adjetivo, então o Normalizador procura em seu dicionário, a forma normal do adjetivo “assado”, como mostrado na figura 24, onde se encontram também as outras formas normais.

Dicionário DELAF		
assados	assado	adjetivo
assados	assado	substantivo
assados	assar	verbo

Figura 24 – Trecho do dicionário utilizado pelo Normalizador

Como dito anteriormente, o Curupira gera um arquivo de saída com a análise feita. Esse arquivo gerado é utilizado pelo Normalizador para buscar no seu dicionário as formas canônicas das palavras.

3.6. Gerador da ConceptNet

Da mesma forma que a original, a ConceptNet é gerada a partir de um processo automático, extraindo as sentenças da base do OMCS brasileira. Esse processo também ocorre em 3 fases:

Fase de extração

As sentenças extraídas através do site OMCS, armazenadas na base, serão transformadas em declarações.

As 10 regras de extração transformam as sentenças em declarações da ConceptNet de forma direta, pois todas essas sentenças se apresentam como atividades com formato semi-estruturado. Todas as atividades do site brasileiro são baseadas em *templates* em que o colaborador deve apenas preencher o espaço em branco. Assim, uma sentença gera diretamente uma declaração. Por exemplo: Se a sentença “Você geralmente encontra um(a) **nhoque** em um(a)___” é preenchido pelo o usuário com “restaurante” é gerada a declaração (LocationOf, ‘nhoque’, ‘___’, ‘f=1; i=0’). Assim, um dos “conceitos” da declaração é o texto digitado pelo usuário, e o outro conceito é “nhoque”, que é um termo recuperado da base, utilizado na retro-alimentação. A tabela 11 lista os *templates* existentes e as declarações geradas a partir deles. Os campos dos

templates sinalizados por () representam campos que são preenchidos automaticamente pelo site através da retro-alimentação. Já os campos sinalizados por [] representam campos que serão preenchidos pelo usuário do site e as abreviações que são encontradas dentro deles correspondem aos tipos de entradas que é esperado. Os significados dessas abreviações são mostrados na tabela 9 da página 41.

Tabela 9 – Declarações geradas diretamente a partir de *templates*

Atividade	Template	Declaração
Coisas	Uma coisa que você pode encontrar em um(a) (L) é um(a) [S]	(LocationOf,S,L)
Sentenças	Você geralmente quer um(a) (S) para [VI]	(UsedFor,S,VI)
Usos	Um(a) (S) é usado para [VI]	(UsedFor,S,VI)
Localização	Você geralmente encontra um(a) (S) em um(a) [L]	(LocationOf,S,L)
	Você geralmente encontra um(a) [S] em um (a) (L)	(LocationOf,S,L)
Pessoas	Pessoas [VP] quando elas [VP]	(MotivationOf,VP,VP)
Analogia	Um outro jeito de dizer (VI) é [VI]	(ConceptuallyRelatedTo,VI,VI)
Problemas	Quando se tenta (VI') um problema encontrado pode ser [VI'']	(ConceptuallyRelatedTo,VI'',VI')
Ajuda	Quando pessoas (VP) uma forma de ajudar é [VI]	(MotivationOf,VI,VP)
Cores	A cor (cor gerada pelo sistema) faz ter vontade de [VI]	(DesirousEffectOf, cor, VI)
	A cor (cor gerada pelo sistema) me lembra um [VI]	(ConceptuallyRelatedTo,cor,VI)
	(S') me faz lembrar a cor [S'']	(ConceptuallyRelatedTo,S'',S')
Saúde	Para (cuidar de alguém doente) em casa deve-se [VI]	(ConceptuallyRelatedTo, cuidar de alguém doente, VI)
	Antes de (começar a cuidar de alguém doente em casa) deve-se [VI]	(PrerequisiteEventOf, começar a cuidar de alguém doente em casa,VI)
	Para poder (cuidar de alguém doente em casa) é preciso ter [S]	(PrerequisiteEventOf, cuidar de alguém doente em casa,S)
	Para poder (cuidar adequadamente de alguém doente em casa) é preciso estar [E]	(PrerequisiteEventOf, cuidar adequadamente de alguém doente em casa,E)
	Para (dar o remédio a alguém doente) deve-se [VI]	(PrerequisiteEventOf, dar o remédio a alguém doente,VI)
	O que leva alguém (cuidar de um doente) em casa é [S]	(MotivationOf, S, cuidar de um doente)
	Uma (pessoa doente) gosta de [VI]	(DesirousEffectOf, pessoa doente,VI)
	Para conseguir estar (E) deve-se [VI]	(PrerequisiteEventOf,E,VI)
	As pessoas (cuidam de um doente em casa) para [VI]	(MotivationOf, cuidam de um doente em casa,VI)
	(S) ajuda no cuidado da pessoa doente porque [J]	(EffectOf,J,S)
É (difícil fácil) (cuidar de alguém doente em casa) porque [J]	(EffectOf, cuidar de alguém doente em casa,J)	

Um problema encontrado nesta fase em que se converte o texto em língua em declarações, foi na atividade “Sentença”. O template da atividade tem o seguinte formato: “Você (geralmente) quer um(a) (S) para [VI]”. No lugar da palavra “geralmente” também é possível aparecer “as vezes”, “freqüentemente”, “muito freqüentemente”, “quase sempre”, “quase nunca”, “raramente” ou “nunca”. O problema ocorre quando aparece uma sentença com “quase nunca”, “raramente” ou “nunca”, pois o usuário acaba completando a sentença com um “conceito”¹⁸ que não está relacionado com o outro “conceito” recuperado do banco de dados. Por exemplo, “Você nunca quer um livro para ___” pode ser preenchido com a palavra “comer”. O Gerador original, cria uma relação UsedFor entre “livro” (termo recuperado do banco de dados) e “comer” (palavra inserida pelo usuário), desconsiderando a palavra “nunca”, que indica que não há relação entre esses conceitos. Para contornar essa questão, no Gerador brasileiro, as sentenças que possuem “quase nunca”, “raramente” ou “nunca” não geram nenhuma declaração. Outros estudos têm sido realizados para solucionar esse problema.

Fase de normalização

As declarações extraídas na fase de extração são normalizadas. Como o Curupira não possui a função de apresentar a forma normal das palavras, um normalizador foi construído para esta função.

O processo de normalização se realiza da seguinte maneira:

1. O Curupira divide a sentença em *tokens* e classifica através das *tags*

Exemplo: Comeram cinco frangos assados → Comeram/**Verbo**
cinco/**Número** frangos/**Substantivo** assados/**Adjetivo**

¹⁸ Conceito: Um nó da rede semântica

2. O Curupira envia a informação ao normalizador. Este procura no seu dicionário, o DELAF (www.nilc.icmc.usp.br:8180/unitex-pb), a forma normal de cada palavra para a flexão específica dada pelo Curupira. Por exemplo: a palavra “assados” pode ter as formas canônicas “assado” se for um adjetivo (ex. Comi cinco frangos **assados**) ou um substantivo (ex. Comi uns **assados**) e “assar” se “assados” estiver sendo usado como um verbo (ex. Os frangos foram **assados** depressa).

Exemplo: Comer**am** cinco frangos assados → Comer cinco frango assado

3. O Normalizador retira artigos e números

Exemplo: Comer **cinco** frango assado → Comer frango assado (Foi retirado “cinco”)

Uma questão observada no Curupira é que em algumas ocasiões um verbo seguido de substantivo não é corretamente analisado. Um exemplo encontrado foi a análise feita no texto “Fazer compras”. O Curupira entendeu que “Fazer” é um verbo, mas “compras” também foi considerado um verbo, e numa análise correta, o programa deveria entender como um substantivo. Para contornar essa questão, foi inserido um tratamento. Assim, quando o Curupira encontra um verbo seguido do outro, se o segundo verbo não estiver no infinitivo (ex. **Fazer comer** rápido não é bom), gerúndio (ex. Ele **andou fazendo** besteiras) ou particípio (ex. Isso **foi resolvido**), ele é considerado um substantivo, como no caso de “Fazer compras”.

Fase de relaxamento

Depois de obter as declarações normalizadas, as declarações passam pelo mesmo conjunto de heurísticas de relaxamento da ConceptNet original. A seguir é apresentada uma breve visão sobre as heurísticas:

1. As declarações duplicadas são retiradas e o número de repetições que existia é contabilizado no campo “frequência”. Assim, como mostrado no exemplo da figura 25, se havia três declarações, passará a existir uma só, com frequência 3.

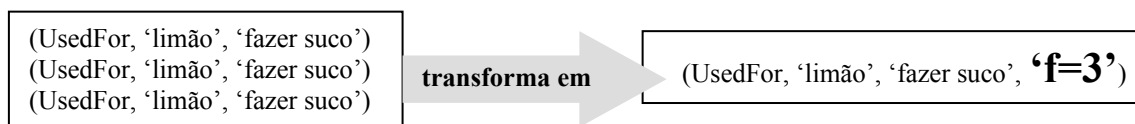


Figura 25 – Tratamento de nós duplicados Definição do valor da frequência

2. Generalizações temáticas e léxicas são produzidas relacionando o conhecimento mais específico ao mais geral, produzindo assim, relações do tipo “SuperThematicKLine”. No exemplo da figura 26 mostra que “comprar” é uma forma genérica de “comprar comida”.

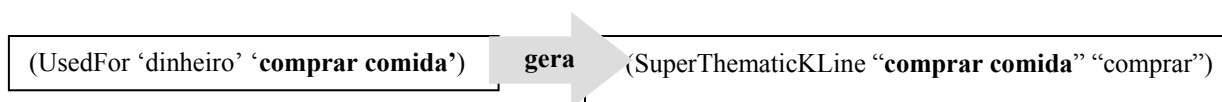


Figura 26 – Generalizações de nós

3. Todas as declarações criadas terão os valores das inferências calculados.

Como não há nenhuma relação ‘IsA’ gerada a partir dos *templates* do site brasileiro, não são geradas as novas declarações com a relação PropertyOf (figura 27), como ocorre no original, visto no capítulo anterior.

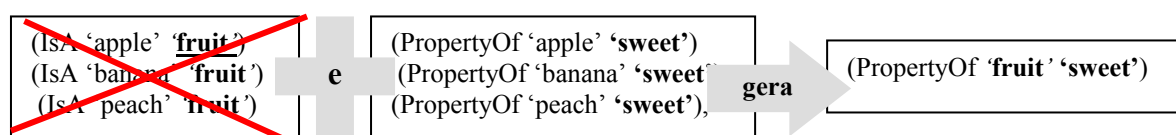


Figura 27 – Inexistência da relação IsA

3.7. A API brasileira do Projeto Open Mind Common Sense

A API desenvolvida neste trabalho é responsável por ajudar as máquinas a utilizar o conhecimento de diferentes modos, para que possam pensar, planejar, explicar, prever e todas as outras habilidades mentais de um humano.

Na prática ela faz a comunicação entre a aplicação computacional com a rede semântica. Quando o usuário da aplicação computacional digita algo, é feita uma análise do texto pelo Curupira e algumas funções da API são chamadas. Através da API, três tipos de inferência na ConceptNet são realizados:

1. Encontrar o contexto

A função `GetContext()` determina o contexto sobre um conceito ou sobre vários conceitos. Por exemplo, quando um usuário digita o fragmento de uma sentença “ir para cama”, a rede semântica retorna os conceitos relacionados que foram encontrados, tais como, “despir”, “ir dormir”, “dormir”, “deitar-se”, “fechar os olhos”, “desligar a luz”, “sonho”, “escovar os dentes”, “roncar”, etc. Esta lista de nós que o `GetContext()` retorna está relacionada à distância entre nós e o número de caminhos.

2. Inferência de encadeamento

A função `FindPathsBetween()` encontra os caminhos entre dois nós. Por exemplo, se é digitado ‘comprar comida’ e ‘dormir’, e esses dois nós existem na rede, a função mostrará todo o caminho percorrido: ‘comprar comida’ → ‘ter comida’ → ‘comer comida’ → ‘sentir-se cheio’ → ‘sentir sono’ → ‘dormir’.

3. Analogia conceitual

A função `GetAnalogousConcepts()` retorna uma lista de conceitos análogos de um conceito. Por exemplo, digitando ‘divã’, a função `GetAnalogousConcepts()` retorna como resultados conceitos com: ‘sofá’, ‘cadeira’, ‘cama’, ‘banco’.

Do mesmo modo, a API trabalha para enviar à aplicação computacional todas as possíveis interpretações e não tem como objetivo apresentar uma única solução certa.

3.8. Considerações Finais

Neste capítulo foi apresentado o projeto Open Mind Common Sense no Brasil, como o site e a rede semântica foram desenvolvidos e como o Curupira, analisador de sentenças da língua portuguesa desenvolvido pelo NILC, (<http://www.nilc.icmc.usp.br/nilc/index.html>) apóia esse sistema.

Foi discutido como surgiu o interesse da equipe do projeto OMCS pela pesquisa no Brasil, depois foi apresentada uma visão geral do sistema criado e nas seções posteriores, uma visão mais detalhada de cada parte do sistema OMCS, bem como as soluções desenvolvidas para o processamento de linguagem natural em português. Também foram citadas algumas pesquisas sendo desenvolvidas pela equipe brasileira com o uso do senso comum.

Foram feitas algumas comparações entre o site brasileiro e o original. Além disso, foram apresentadas quais são as atividades do site brasileiro e quais foram os dados pessoais dos colaboradores colhidos. Com essas informações pretende-se gerar sistemas que ajudem pessoas de diferentes perfis a se comunicarem. Também foi apresentado funcionamento do esquema de retro-alimentação de dados para geração de novas sentenças. E os meios utilizados para atrair colaboradores também são mencionados.

Neste trabalho, foram feitas algumas modificações no Gerador da rede semântica e na API, para que eles passassem a chamar as funções da nova classe construída para fazer a comunicação com o Curupira. Também foi preciso desenvolver

um Normalizador, já que o Curupira não oferecia essa funcionalidade Além disso, foi necessário converter o site da língua inglesa para a portuguesa, e novas atividades foram inseridas. Depois foi detalhado mais sobre o que é o Curupira e como ele ajuda na geração da ConceptNet. Observando que todas as declarações são geradas diretamente dos templates.

Para que fosse possível realizar a adaptação, foi preciso trabalhar com as linguagens de programação envolvidas, como o JSP para tradução e inserção de novas atividades no site e o Python para trabalhar com a língua portuguesa e substituir o MontyLingua pelo Curupira para criar e usar a ConceptNet. Além disso, foi necessário entender questões relacionadas com o processamento de língua natural e buscar auxílio dos pesquisadores envolvidos mais diretamente com esta área.

Porém, realizar estudos sobre o senso comum para fornecer tal conhecimento às máquinas no Brasil ainda é um campo de pesquisa novo, sendo uma grande oportunidade de colaborar com pesquisas das áreas de Inteligência Artificial (IA), Interação Humano-Computador (IHC) e Interação Humano-Humano (IHH). No capítulo seguinte serão apresentadas algumas das potencialidades da utilização de senso comum em aplicações computacionais, e desse modo, tentando colaborar com as pesquisas das áreas citadas.

CAPÍTULO 4 - SENSO COMUM PARA RECONHECIMENTO DE DIFERENÇAS CULTURAIS

4.1. Considerações Iniciais

O senso comum, como definido no capítulo 2, está relacionado com aos aspectos especiais, físicos, sociais, temporais e psicológicos vividos durante o cotidiano (SINGH, 2002A). Isso significa que a noção de senso comum é adquirida através da interação com o lugar em que se vive, desse modo, uma pessoa que mora em um determinado lugar tem a percepção de senso comum diferente de uma pessoa que reside em um outro. É possível dizer que o senso comum muda com o ambiente, pois a percepção de senso comum também muda. Assim:

O senso comum pode mudar com o tempo: até há pouco tempo, era de senso comum que para “abrir a porta” era preciso girar a maçaneta e empurrar a porta. Hoje em dia, é comum encontrar em diversos estabelecimentos comerciais as portas automáticas, tendo mudado o senso comum de “abrir a porta”, pois neste caso, é preciso apenas andar em direção à porta e ser percebido pelos sensores. Um outro exemplo é o senso comum para “utilizar um telefone público”. Antes as pessoas deveriam ter uma ficha a mão, hoje só se usa cartão.

O senso comum pode mudar com o lugar: para as pessoas que moram na zona rural, é de senso comum considerarem uma carroça como um meio de transporte, fato inaceitável para os moradores das grandes cidades, que usam ônibus, metrô, carro e helicóptero para se locomoverem, enquanto os que vivem no deserto, usam um camelo.

A partir dessas constatações, acredita-se que o conhecimento de senso comum pode ajudar na identificação de diferenças existentes entre as culturas. Para isso,

a base de fatos coletados através do site do projeto Open Mind Common Sense (OMCS) pode ser utilizada para ajudar a mostrar tais diferenças.

Em relação à concepção da palavra “cultura”, segundo o dicionário Aurélio (DICIONÁRIO AURÉLIO, 1995), cultura é um padrão de comportamento, das crenças, das instituições e de outros valores espirituais e materiais transmitidos coletivamente e característicos de um grupo social. Um grupo social pode ser classificado através dos mais variados critérios, por exemplo: sexo, faixa etária, religião, lugar que residem (países, estados, cidades, etc), preferências (roqueiros, motoqueiros, *skatistas*, hippies, etc), entre outros. As diferenças culturais expressam a “visão do mundo” que um grupo de pessoas têm.

Ao se reconhecer os aspectos culturais refletidos nos fatos armazenados na base do projeto OMCS, pôde-se identificar o potencial de usos dessa informação para algumas aplicações computacionais.

Neste capítulo será discutido sobre o uso de senso comum para reconhecer diferenças culturais. Na próxima seção será mostrado como o uso de informações referentes às culturais pode ajudar a projetar interfaces adequadas para diferentes perfis e também pode melhorar os resultados de uma busca para recuperação de informações. Ainda neste mesmo capítulo, será discutido como as características culturais podem ser identificadas e classificadas segundo alguns pesquisadores do assunto e como elas podem ser utilizadas em aplicações computacionais. E por fim, na última seção serão mostradas as considerações finais do capítulo.

4.2. Importância do Conhecimento sobre Cultura em aplicações computacionais

Acredita-se que as diferenças culturais identificadas nos fatos de senso comum podem ser utilizadas para:

- Projetar interfaces de aplicações computacionais mais adequadas para um certo tipo de usuário questão relacionada à área de Interação Humano-Computador (IHC).
- Recuperar informações relevantes, questão relacionada à área de Recuperação de Informação (RI).
- Ajudar as pessoas a se comunicarem, questão relacionada à área de Interação Humano-Humano (IHH).

Para ajudar na identificação das características culturais que podem ser consideradas durante a construção de uma dessas aplicações computacionais, alguns pesquisadores definiram atributos que podem ser utilizados para ajudar a classificar culturas (MACGREGOR, HSIEH E KRUCHTEN, 2005), tais como:

Kluckhohn & Strodtbeck (2000): definiram 5 atributos e cada um deles tem 3 valores que classificam uma cultura.

- Natureza humana: (1) Boa, (2) má ou (3) uma mistura de bondade e maldade.
- Relação com a natureza: (1) Submissa à natureza, (2) dominadora da natureza ou (3) harmônica com a natureza.
- Senso de tempo: (1) Prioriza costumes tradicionais, (2) planos futuros ou (3) eventos atuais.

- Relação social: (1) Organizada numa hierarquia linear, (2) possui interesses coletivos ou (3) interesses individuais.
- Espaço: (1) A vida é conduzida publicamente, (2) de forma privada ou (3) ora de forma privada e ora publicamente.

Geert Hofstede (1997): fez um estudo para identificar aspectos culturais entrevistando empregados da IBM de 53 países. Ele identificou 5 atributos com valores de 0 a 100 que classificam as culturas:

- Poder: aceitação da diferença entre líderes e seguidores
- Individualismo/ Coletivismo: prioriza grupo ou a si próprio
- Masculino/ Feminino: os papéis de cada um são iguais ou distintos
- Fuga da incerteza: tolerância a situações desconhecidas e ambíguas.
- Orientação de período longo/ curto: comprometidos e lentos para realização das atividades ou rápidos e sem muito comprometimento para executar tarefas.

Edward Hall (1976): examina a cultura em relação ao tempo e ao padrão de comunicação:

- Tempo: uma cultura pode ser: (1) mono-crônico: uma tarefa é realizada de cada vez, isto é, uma tarefa deve ser finalizada antes de começar uma outra; ou (2) poli-crônico: várias tarefas são realizadas simultaneamente e interrupções são frequentes.
- Padrão de comunicação: (1) transmissão de mensagens mais explicitamente com intuito de evitar ambigüidades ou (2) mensagens são ditas em poucas

palavras, assumindo que os participantes da conversa estão contextualizados.

Alguns exemplos de uso desses atributos e como as características culturais identificadas podem ser utilizadas para ajudar usuários ou desenvolvedores de sistemas computacionais a realizarem suas tarefas serão descritos nas próximas seções.

1. Importância do Conhecimento sobre Cultura para o Projeto de Interfaces

Como os usuários de culturas diferentes usam a Internet para propósitos diferentes, é preciso mais do que apenas fornecer interfaces em versões de diferentes línguas (CHAU ET AL, 2002). A interface deve conter elementos que sejam agradáveis ao usuário (CHAU ET AL, 2002).

Segundo Borgam (1992), culturas variam em estilo de aprendizagem, orientação espacial, preferência de cores e uso da linguagem. Por isso, projetar interfaces para agradar pessoas de uma cultura em que não se conhece é uma tarefa bastante difícil (MARCUS, 2002).

Mais do que apenas traduzir palavras, a noção de preferências culturais deve ser considerada no projeto de interfaces (SUN, 2001). Segundo Huatong Sun (2001), esse processo é realizado em dois níveis:

1. Ajuste de características do software, incluindo tradução, datas, pesos, endereços, moeda, etc. para refletir as convenções e as necessidades do público alvo;
2. Ajuste da estética, imagens, cores, lógica, padrões de funcionalidade e comunicação conforme o público alvo.

Sun (2001) foca seu trabalho em quatro principais categorias: língua, visual (figuras e gráficos), cores e layout da página (simétrico, assimétrico, centrado, etc.).

Existem pesquisas, tais como (MARCUS E GOULD, 2000; BORGAM, 1992; MARCUS, 2002) sendo feitas sobre como entender as características de uma cultura e como elas podem afetar o projeto de interface. Atributos como atração, dinamismo, fé, intenções, localização, preferências, etc, têm importância diferente dependendo da cultura (BAILEY, GURAK, E KONSTAN, 2001). A seguir, são apresentados alguns exemplos de como os atributos que definem características culturais podem ser utilizados no projeto de interfaces:

Kluckhohn & Strodtbeck:

- Natureza humana: Está relacionado com o gosto de uma cultura por uma interface mais extravagante ou mais serena.
- Relação com a natureza: Pode determinar o uso ou não de elementos representantes da natureza na interface.
- Relação com o tempo: Pode determinar o uso de elementos tradicionais ou modernos na interface.
- Relação social: Pode determinar o uso de elementos que mostram coletivismo (por exemplo, fotos de grupos de pessoas) ou individualismo (por exemplo, fotos de uma única pessoa).
- Espaço: Pode determinar o uso de elementos que mostram informações particulares

Geert Hofstede: Segundo (MARCUS E GOULD, 2000), os atributos podem influenciar nos seguintes aspectos do projeto de interface:

- Poder: Acesso à informação (bem ou mal estruturado); uso de símbolos religiosos e de nacionalismo; uso de selos, brasões, e certificações que

mostram autoridade; importância dada aos líderes ou aos subordinados, clientes e empregados; preocupação com segurança e restrições ao acesso de informações ou liberdade ao usuário. A figura 28 e 29 apresentam exemplos de interfaces com este atributo.

- Individualismo/ Coletivismo: Uso de imagens de sucesso que demonstram materialismo e consumismo ou imagens de eventos políticos-sociais;
- Masculino/ Feminino: Em culturas que possuem um alto valor do atributo “masculino” tendem a fazer distinções entre interfaces para diferentes idades e sexos; resultados rápidos e tarefas limitadas; navegação orientada e controlada; uso de jogos e competições para atrair atenção; uso de gráficos, sons e animações. Já em culturas femininas as pessoas de sexos diferentes teriam uma mesma interface; ao invés de competições, existe cooperação; para atrair a atenção são usados elementos visuais estéticos.
- Fuga da incerteza: Em culturas com alto grau de fuga da incerteza, as interfaces apresentam simplicidade, com escolhas limitadas e uma pequena quantidade de dados; tenta-se prever os resultados de uma ação antes dela ser realizada; tenta-se reduzir erros que podem ser cometidos pelos usuários; inúmeras dicas (cor, sons, etc.) são usadas para reduzir ambigüidades. Já em culturas com baixo grau de fuga da incerteza existem diversas opções de escolhas e uma grande quantidade de conteúdo; incentivo para que os usuários explorem e arrisquem; menor controle da navegação, como por exemplo, *links* podem abrir novas janelas fazendo desviar da localização original.

- Orientação de período longo/curto: a interface permite que os objetivos sejam alcançados rapidamente ou requer mais passos.



Simetria axial, foco no brasão oficial da universidade, foto de líderes e monumentos mostrando que as pessoas exercem um papel pequeno

Figura 28 – Alto valor do atributo “Poder” – site de uma universidade da Malásia



Ênfase nos estudantes (não líderes), fotos de pessoas de ambos os sexos.

Figura 29 – Baixo valor do atributo “Poder” – site de uma universidade da Holanda

Marcus (2002) menciona como os aspectos culturais podem atuar a percepção dos usuários de sistemas interativos no projeto de interfaces. Ele acredita que sistemas considerando diferenças culturais podem, entre outras coisas: (1) influenciar na noção de usabilidade das pessoas; (2) persuadir e estabelecer confiança dos usuários; (3) possibilitar novas percepções e (4) mudar a forma de navegação.

É uma questão complexa analisar as necessidades, desejos e expectativas de culturas diferentes (MARCUS E GOULD, 2000). Metáforas, navegações, interações e

aparência podem ser confusas e até ofensivas para usuários de uma certa cultura (MARCUS E GOULD, 2000). Os usuários de culturas diferentes têm diferentes maneiras de tomar decisões, o que pode influenciar nas suas preferências. Por exemplo, se uma pessoa planeja viajar de avião, ela pode preferir encontrar primeiro a tabela de horários ou, então, encontrar primeiro as informações sobre a credibilidade da companhia aérea..

Um exemplo que mostra como as questões culturais podem ser essenciais é a constatação citada por Marcus (2002). Ele menciona em seu artigo que os japoneses preferem softwares desenvolvidos por europeus, mais especificamente escandinavos, do que os softwares desenvolvidos por americanos. Para eles, os softwares europeus aparentam mais elegância e sensibilidade. Já os softwares americanos são menos educados para os gostos japoneses.

Huatong Sun (2001) em seu artigo, cita alguns exemplos para demonstrar a importância dos aspectos culturais nos projetos de interface:

- Nos elementos da interface: Os usuários brasileiros gostam de cores mais vibrantes e páginas com muitas figuras. Os alemães sentem-se satisfeitos com links organizados em ordem alfabética.
- Símbolo cultural: Os brasileiros e os chineses se sentem confortáveis quando vêem figuras de suas culturas (como o Pão de Açúcar e a flor de lótus, uma flor popular na China).
- Modos de mostrar símbolos culturais: os alemães preferem componentes textuais, já os brasileiros e os chineses preferem componentes visuais coloridos.

Apesar da importância dessas questões levantadas, alguns desenvolvedores ainda encontram uma grande dificuldade de obter apoio para realizar pesquisas a

respeito da cultura dos usuários que irão interagir com a interface projetada (MARCUS, 2002).

Além disso, considerar aspectos culturais nos projetos de interfaces não é uma tarefa fácil . Já que as crenças, atitudes e valores de um grupo mudam com o tempo. O trabalho de coletar os dados que revelam as características culturais, e disponibilizar para todos que gostariam de desenvolver interfaces considerando tais diferenças, pode ser caro e cansativo.

Assim, acredita-se que o projeto OMCS pode ajudar nesse objetivo. Os fatos para identificar o senso comum estão sendo coletados continuamente, já que a idéia de “senso comum” de uma determinada cultura muda com o tempo. Além disso, aquisição das informações sobre as culturas através da colaboração das pessoas através do uso da Internet não gera altos custos. E desenvolvedores podem acessar a base que armazena fatos de senso comum sem custos e podem utilizar esses dados em seus projetos de interface.

2. Recuperação de Informação Considerando as Diferenças Culturais

A quantidade de páginas Web disponibilizada na Internet cresce o tempo todo e conseqüentemente encontrar informações relevantes torna-se uma tarefa cada vez mais difícil (BIRUKOV, BLANZIERI E GIORGINI, 2005). Contudo, quando se considera uma comunidade com interesses comuns, é possível melhorar a qualidade dos resultados de uma busca, através da utilização do conhecimento extraído da base de dados de fatos de senso comum. Com essa base, é possível identificar os comportamentos de pessoas de uma mesma cultura. Assim, quando um usuário faz uma busca, os resultados são exibidos de acordo com as observações do comportamento de outros usuários da mesma cultura que realizaram a mesma busca. Pessoas de culturas diferentes podem esperar respostas diferentes para uma mesma busca.

Spink (2002) fez um estudo comparativo mostrando as diferenças entre as pesquisas realizadas na Web por americanos e europeus. O resultado mostra que há algumas diferenças nos tópicos pesquisados e no comportamento dos usuários americanos e europeus. Por exemplo, os usuários americanos são mais focados em pesquisas relacionadas a negócios enquanto os ingleses preferem ler sobre relações sociais.

Essas diferenças culturais e sociais representam um grande desafio para se construir uma máquina de busca eficiente.

A base de senso comum pode ser utilizada por máquinas de busca para refletir as diferenças culturais e mostrar resultados mais relevantes aos usuários de uma determinada cultura. No trabalho de Faaborg e Lieberman (2006), eles apresentam um *browser* que é capaz de gerar hipertexto de acordo com o contexto detectando os possíveis objetivos do usuário. Segundo eles, a utilização de senso comum melhora a capacidade dos computadores entenderem a intenção do usuário com mais facilidade. Eles mencionam também que através do uso de redes semânticas, o computador pode generalizar informações e antecipar as ações dos usuários. Assim, conhecendo as características culturais dos usuários, os hipertextos gerados podem ser mais adequados aos interesses deles.

Em relação às questões de Interação Humano-Humano (IHH), mencionadas anteriormente, serão apresentados os principais pontos no próximo capítulo.

4.3. Considerações Finais

Neste capítulo foi apresentado como o conhecimento de senso comum pode ajudar na identificação de diferenças existentes entre as culturas. Também foram exemplificadas algumas aplicações computacionais que podem ser auxiliadas com o

reconhecimento das características culturais dos usuários. Além disso, foram apresentadas algumas pesquisas feitas sobre o tema para entender as características de uma cultura.

Identificar diferenças culturais é um grande desafio, já que normalmente o conhecimento cultural é implícito e é difícil perceber o que é dependente culturalmente.

Ainda foi visto que um melhor entendimento dos grupos culturais e suas razões para usarem um determinado programa pode ajudar a melhorar o próprio software.

Vale mencionar que neste trabalho os aspectos culturais ainda não foram aplicados na prática em todas as suas potencialidades apresentadas, mas foram realizadas algumas análises que indicam a importância e o potencial de uso de senso comum na consideração das questões culturais quando se pensa em sistemas iterativos.

Assim, neste capítulo foi discutido sobre o uso de senso comum para identificar diferenças culturais. No próximo capítulo deste trabalho será apresentado o uso dessas informações relacionadas às diferenças culturais para ajudar na comunicação entre as pessoas.

CAPÍTULO 5 - USANDO SENSO COMUM PARA FACILITAR A COMUNICAÇÃO ENTRE PESSOAS MOSTRANDO AS DIFERENÇAS CULTURAIS

5.1. Considerações Iniciais

A comunicação através da Internet é cada vez mais comum em nossas vidas. Hoje em dia as pessoas gastam um certo tempo do seu dia em frente ao computador para se comunicarem. Programas de e-mail, sala de bate-papo e programas para envio de mensagens instantâneas são alguns exemplos de ferramentas utilizadas para esse fim. As conversas podem acontecer entre pessoas que se conhecem ou entre aquelas que se encontraram casualmente, podendo ser moradoras de cidades ou até mesmo de países diferentes. É comum as pessoas se encontrarem pela primeira vez por meio de uma interface de um computador (ISBISTER ET AL., 2000). Esses ambientes acabam possibilitando uma maior interação à distância, e conseqüentemente através de culturas (SETLOCK, FUSSELL E NEUWIRTH, 2004).

Neste cenário, através da comunicação intercultural, as pessoas acabam trocando experiências culturais, o que traz grandes benefícios e desafios bastante interessantes. Para que isso seja possível, é necessário entender as características específicas sobre as culturas envolvidas (ZORN, 2005), pois diferentes culturas têm diferentes noções de como começar e como encaminhar uma conversa (ISBISTER ET AL., 2000).

Devido às diferenças nas regras locais e culturas distintas (SETLOCK, FUSSELL E NEUWIRTH, 2004), as informações culturais poderiam ser de grande ajuda aos participantes da conversa. Assim, a comunicação poderia fluir de forma mais fácil e os participantes poderiam se sentir mais confortáveis e se sentirem melhores com a

interação, e conseqüentemente, poderiam formar impressões mais positivas a respeito de outras culturas (ISBISTER ET AL., 2000).

Neste capítulo é mostrado o potencial do uso da base de conhecimento de senso comum para facilitar a comunicação entre as pessoas, através do fornecimento de mais informações sobre outras culturas. Acredita-se que o senso comum pode fornecer algumas diretrizes sobre diferenças culturais em aplicações computacionais para assim dar um suporte melhor à interação humano-humano (IHH) em todo o mundo.

Diferentes culturas têm diferentes noções de como começar e como desenvolver uma conversa. O que é um assunto inofensivo em uma cultura pode ser inadequado em outra. (BROWN E LEVINSON, 1987).

Diversos atributos, como a de Kluckhohn e Strodtbeck, Hall e Hofstede, mencionados no capítulo anterior, podem ajudar a identificar as características culturais. Conhecendo tais características, pode se tentar realizar uma comunicação mais agradável para os participantes da conversa. A seguir, são apresentados alguns exemplos de como esses atributos podem ser utilizados para entender uma cultura e seu estilo de comunicação:

Kluckhohn e Strodtbeck:

- Natureza humana (boa, má ou ambos): Em culturas consideradas de natureza boa, às vezes, uma brincadeira pode causar constrangimento.
- Relação com a natureza (submissa, dominadora ou harmônica): Pode indicar a maneira como os assuntos relacionados com conservação da natureza, animais, poluição, etc. podem ser tratados.

- Relação com o tempo (costumes tradicionais, planos futuros ou eventos atuais): As pessoas de uma cultura mais tradicional geralmente não gostam de utilizar gírias e alguns assuntos mais pessoais podem constrangê-las.
- Espaço (A vida é conduzida publicamente, de forma privada ou ambos): Pode mostrar que as pessoas de uma cultura não se importam em falar de assuntos particulares.

Edward Hall:

- Tempo: (1) mono-crônico: um assunto deve ser tratado de cada vez e fazer interrupções podem ser considerado deselegante; (2) poli-crônico: vários assuntos podem ser tratados simultaneamente e com interrupções frequentes.
- Padrão de comunicação: (1) transmissão de mensagens é feita de maneira mais explícita com intuito de evitar ambigüidades ou (2) mensagens são ditas em poucas palavras, assumindo que os participantes da conversa estão contextualizados.

Além dos atributos mencionados, características lingüísticas também tornam distintos os estilos de comunicação entre as culturas, tais como o “*Politeness*” e a “Linguagem inclusiva” (*Inclusive Language*) (SETLOCK, FUSSELL E NEUWIRTH, 2004).

- *Politeness*: Fatores considerados cortês ou educado. Por exemplo, em algumas culturas, é apropriado perguntar sobre o tipo de comida que será servida numa festa, enquanto em outra isso pode ser considerado muito deselegante.
- Linguagem inclusiva: É a maneira como as pessoas se identificam. Elas podem se identificar como indivíduos ou como membros de um grupo. Essa

característica pode-se refletir, por exemplo, no maior uso do pronome “nós” ou “eu” para se identificar.

Neste capítulo, além dos atributos para reconhecer algumas características de culturas, na próxima seção será apresentada uma ferramenta de comunicação que utiliza o conhecimento de senso comum para mostrar as diferenças culturais às pessoas envolvidas na comunicação. A última seção, as considerações finais do capítulo são apresentadas

5.2. Estudo de Caso – Análise dos hábitos alimentares

Ferramentas de comunicação (e-mail, chat, fórum, etc.) oferecem um potencial de comunicação internacional e intercultural, que pode atrair e unir pessoas de diferentes culturas e regiões, no entanto ainda existem restrições. Essas ferramentas de comunicação parecem ter um grande potencial para trocas de informações de forma democrática, para aprendizado colaborativo, comunicação intercultural, fluxo de informação, etc. (ZORN,2005).

Segundo Zorn (2005), a diversidade cultural promove grandes benefícios e desafios. Quando uma pessoa pertence a uma determinada cultura, certos estilos, procedimentos para comunicação, interação e valores são esperados, e assim, torna-se necessário conhecer as características da outra cultura. Como Brown e Levinson (1987) apontam, interações sociais são repletas de ambigüidades, especialmente para o entendimento das crenças, intenções e processos mentais de uma outra cultura.

Em uma comunicação entre pessoas de culturas diferentes, normalmente é definida uma língua a ser utilizada, assim, um ou mais participantes da conversa acabam usando uma língua não nativa. Mesmo tentando se adequar à outra cultura, pesquisas

mostram que as pessoas ainda utilizam o estilo da sua própria cultura (SETLOCK, FUSSELL E NEUWIRTH, 2004).

As culturas também se diferem em relação ao estilo de comunicação. Os indivíduos se comunicam com os outros através da interpretação das mensagens baseados em suas crenças, atitudes e valores. Certas características lingüísticas variam de cultura para cultura, como o exemplo mencionado anteriormente, o *Politenes*.

Devido às diferenças das normas culturais, os participantes podem sentir mais incomodados interagindo com pessoas de culturas diferentes do que em interações na qual eles não pertencem.

Neste trabalho, tenta-se mostrar que o fato de dar algum senso comum aos computadores pode ajudar a minimizar tais problemas, através da apresentação das diferenças culturais durante uma interação entre pessoas.

Com isso, uma versão brasileira do protótipo da aplicação “What is he thinking” (WIHT) (ESPINOSA, LIEBERMAN, 2005) foi construída neste trabalho. WIHT é um programa que fornece sugestões referentes ao tema tratado na interação. Ele observa continuamente o que o usuário está digitando e faz comentários sobre as diferenças culturais, ajudando a evitar possíveis mal entendidos.

O foco do protótipo é promover interação social entre as pessoas no contexto de hábitos alimentares. Este domínio sobre comida foi escolhido para exemplificar o potencial do uso deste tipo de aplicação computacional, mas pode ser evoluído para outros domínios.

A interface do sistema desenvolvido em português, inspirado no WIHT americano, apresenta três seções, como se pode ser observado na figura 30. Na primeira seção são encontrados os campos para escrever o assunto, o endereço de e-mail do

remetente e do destinatário. A segunda seção é onde são mostrados os comentários sobre as diferenças culturais e a terceira parte é o corpo da mensagem. A segunda seção é subdividida em duas seções que mostram como são as características em cada cultura no domínio considerado.

No exemplo do funcionamento do programa mostrado na figura 30, são apresentadas informações referentes aos hábitos alimentares dos brasileiros e dos americanos. Para que o programa possa apresentar tais informações, o sistema utiliza as redes semânticas de cada um desses países, Brasil e Estados Unidos. Essas redes foram criadas a partir dos fatos coletados através dos sites do projeto Open Mind Common Sense (OMCS) dos respectivos países, as bases que contêm fatos específicos de cada cultura. Elas possuem conhecimento que representam as crenças, costumes e valores de brasileiros e americanos.

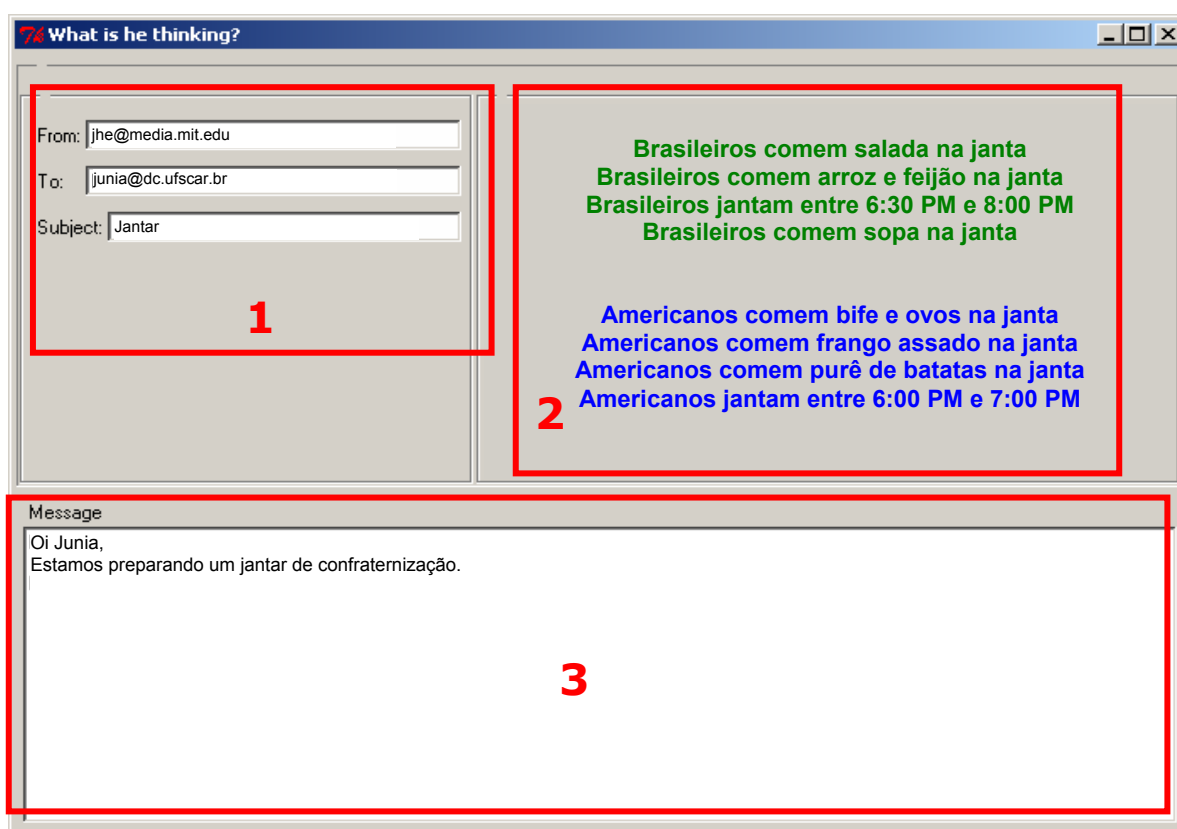


Figura 30 – What is He Thinking (WIHT)

Como as declarações geradas dos fatos da base de dados americana estavam escritas em inglês, neste trabalho, foi traduzido para o português. Feito isso, digitando textos em português, informações sobre as culturas brasileira e americana (bases selecionadas previamente) serão apresentadas, já que os nós da rede semântica americana agora estão em português.

Assim, quando por exemplo, o usuário escreve “Estamos preparando um jantar”, o programa recupera informações relevantes do texto digitado pelo usuário e converte essas informações num formato que o computador possa utilizar.

Depois, uma pesquisa é feita nas duas redes semânticas, a brasileira e a americana. Esse trabalho é feito pela função `GetContext()` da API (descrito com mais detalhes no capítulo 2) em conjunto com o analisador de língua portuguesa do Brasil, o Curupira, que recuperam os nós relacionados. Por exemplos, a palavra “jantar” foi considerada relevante, então os nós relacionados com a palavra “jantar” são recuperados das redes semânticas. Os nós recuperados ajudam a formar as frases listadas na seção 2 do programa WIHT.

Além da comparação feita considerando dois países diferentes, a utilização do conhecimento de senso comum para mostrar diferenças culturais pode ser considerada para facilitar a comunicação entre quaisquer diferentes culturas¹⁹. Assim, o recurso apresentado no programa WIHT pode ser utilizado para ajudar na comunicação entre jovens e adultos, mulheres e homens, mineiros e paulistas, etc.

A figura 31 mostra um exemplo de como o programa WIHT poderia considerar características culturais existentes além das diferenças entre dois países, através do uso dos dados cadastrais dos colaboradores do site OMCS.

¹⁹ Cultura, como definido ao capítulo 2, refere-se à grupos sociais que podem ser classificados através dos mais variados critérios, por exemplo: sexo, faixa etária, religião, etc.

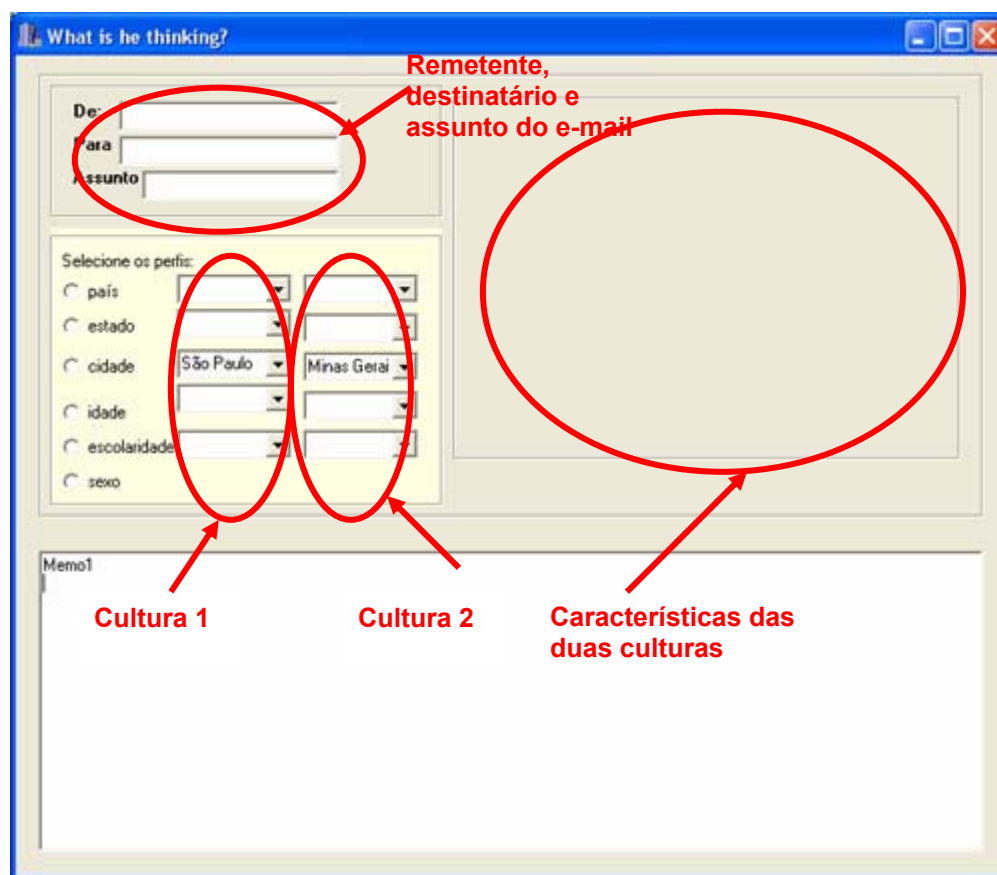


Figura 31 – A versão brasileira do WIHT, com novas funcionalidades

Numa situação em que se um senhor de idade tenta estabelecer uma comunicação com um jovem, esse programa funciona da seguinte maneira (figura 32):

1. Para que o programa execute as funções propostas, primeiramente diversas redes semânticas são criadas pelo Gerador, correspondentes aos perfis existentes dos colaboradores do site OMCS, tais como, uma rede semântica criada a partir dos dados de mulheres, uma outra a partir dos dados dos homens, outra com os dados de pessoas até 13 anos, outra com dados de pessoas de 14 a 18 anos, etc.
2. Com essas redes disponíveis para o uso, o usuário pode começar a utilizar o programa WIHT. Primeiramente, ele seleciona os perfis

desejados (“Cultura 1” e “Cultura 2” da figura 31), no caso citado, o tal senhor selecionaria duas faixas etárias, uma referente à faixa que ele pertence e a outra referente à faixa etária do jovem.

3. Da mesma maneira que o WIHT original, o programa observa continuamente o que o usuário está digitando e faz comentários sobre as diferenças culturais existentes entre as pessoas das faixas selecionadas. Para realizar esse processo, o texto digitado pelo usuário é enviado para a API. Esta por sua vez, com o auxílio do Curupira, converte o texto em um formato utilizado pelo computador. A API então procura as informações somente nas redes semânticas referentes às faixas etárias selecionadas e então, retorna os nós relacionados com o texto digitado.
4. Com os nós obtidos, a API envia à aplicação algumas informações para que o senhor possa interagir com o jovem.

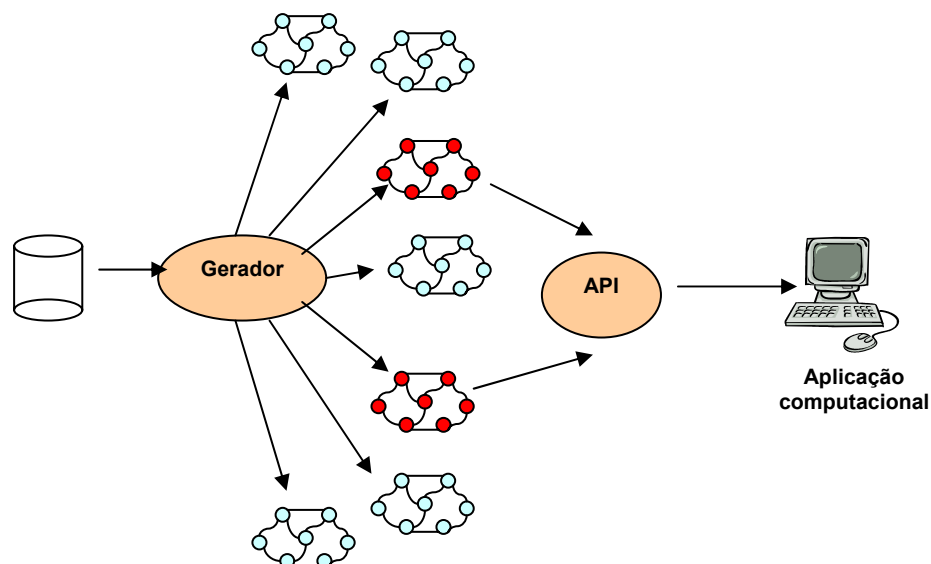


Figura 32 – ConceptNets de cada cultura geradas a partir dos perfis dos colaboradores

5.3. Considerações Finais

Neste capítulo foi discutido que diferentes culturas têm diferentes noções de como começar e como encaminhar uma conversa.

Também foi mostrado que para desenvolver uma comunicação intercultural é necessário mais do que apenas traduzir diferentes linguagens, mas também é necessário se preocupar com crenças, atitude e valores.

Foram apresentados alguns atributos e características de linguagem que podem influenciar na qualidade da interação dos usuários.

Feito este estudo relacionando senso comum e as características culturais, um protótipo que utiliza os recursos do senso comum foi construído neste trabalho para apresentar aos usuários as diferenças existentes entre as pessoas de diferentes países, estados, cidades, faixas etárias, sexos, graus de instrução ou qualquer outra diferença cultural, no contexto de hábitos alimentares. No próximo e último capítulo serão discutidos sobre os resultados obtidos com este trabalho realizado.

CAPÍTULO 6 - RESULTADOS E DISCUSSÃO

6.1. Considerações Iniciais

Conhecimento cultural está associado aos valores e comportamentos que são compartilhados por um grupo (MACGREGOR, HSIEH AND KRUCHTEN 2005). Estudiosos acreditam que os significados compartilhados por uma sociedade são específicos de uma determinada cultura (HOFSTEDE 1991; MESQUITA AND FRIJDA 1992). O significados podem ser influenciados pela linguagem, proximidades geográfica, costumes compartilhados pelas pessoas que vivem dentro de um mesmo ambiente (HOFSTEDE 1980).

Neste trabalho foram utilizadas as bases de conhecimento americana, mexicana e brasileira. A comparação entre essas bases possibilita a descoberta automática das diferenças entre elas (ANACLETO ET AL, 2006A). Vale mencionar que o objetivo do trabalho não é identificar uma descrição completa de uma cultura, mas tem o intuito de reconhecer algumas de suas características que possam ajudar o usuário.

Assim, neste capítulo são mostrados se os dados analisados das três bases, e verificadas as diferenças culturais encontradas. Na seção 6.2 serão mostradas como as análises foram feitas e serão discutidos os dados analisados. Na seção 6.3 será verificada se os objetivos propostos foram atingidos e na seção 6.4 serão apresentadas as considerações finais.

6.2. Discussão do Estudo de caso - Senso Comum e os Hábitos Alimentares

Os dados a seguir dão um reflexo do que pode ser comparado entre as três bases dos países estudados: Brasil, Estados Unidos e México. A idéia não é apresentar uma pesquisa científica definitiva sobre o que os brasileiros, mexicanos e americanos

realmente comem, mas sim, mostrar como uma máquina pode exibir os dados de senso comum dos membros de uma cultura. Vale ressaltar que esses dados nem sempre são completamente verdadeiros, pois eles refletem aquilo que as pessoas de uma cultura acreditam. Também, muitas pessoas aprendem sobre suas culturas de maneira informal.

O primeiro passo foi selecionar na base de dados de cada país os fatos relacionados à comida. Foi possível encontrar fatos relacionados ao horário em que as pessoas têm suas refeições, o que comem em cada uma das refeições, tipos de comidas, ingredientes usados para fazer alguns tipos de alimento e as comidas para ocasiões especiais como festas de aniversário e Natal, como os exemplos exibidos na figura 33:

Bem vindo Marie Tsutsumi! 283 aceitos / 0 em revisão

Busca: Open Mind [Atualize seus dados](#) [Revisão!](#) [Outras at](#)

Busca Resultados para janta

Conhecimento

- Pessoas jantam quando elas estão com fome
- Pessoas jantam entre 19h e 20h30 quando elas estão em um dia normal
- Pessoas comem geralmente o resto da comida do almoço quando elas jantam
- Pessoas pedem chocolate como sobremesa quando elas acabam de jantar
- Pessoas jantam entre 19:00 e 19:30 quando elas estão em um dia normal
- Pessoas pessoas jantam entre 18:00hs e 19:00hs quando elas elas estão em um dia normal
- Pessoas comem alface quando elas jantam
- Pessoas pedem chocolate como sobremesa quando elas acabam de jantar
- Pessoas comem arroz com carne quando elas jantam
- Pessoas jantam entre 18:00 e 21:00 quando elas estão em um dia normal
- Pessoas comem lanche quando elas jantam
- Pessoas pedem sorvete como sobremesa quando elas acabam de jantar
- Você geralmente encontra um(a) esposa em um(a) sala de jantar
- Uma coisa que você pode encontrar em um(a) sala de jantar é um(a) mesa
- Pessoas jantam entre 18:30 e 20:00 quando elas estao em um dia normal

Figura 33 – Fatos relacionados à comida e hábitos alimentares no Brasil

Analisando as categorias identificadas, foram registrados alguns aspectos que se acredita estarem relacionados com valores culturais.

Horário das refeições

Um dos temas sobre comida que normalmente aparece nas bases de conhecimento é o horário das refeições. A tabela 12 mostra os intervalos de horários encontrados na base inseridos pela maioria dos colaboradores.

Tabela 10 – Horário das refeições

	Brasil	México	EUA
Almoço	11:30 a 13:00	14:00 a 16:00	12:00 a 14:00
Janta	18:30 a 20:00	20:00 a 21:00	18:00 a 19:00

Pôde-se observar que as refeições no Brasil e nos Estados Unidos as refeições ocorrem em horários parecidos. Já no México ocorrem um pouco mais tarde, com o almoço e o jantar em torno de duas horas mais tarde.

Tipos de comida

Além da abordagem aos horários das refeições, muitos fatos sobre os tipos de comida são freqüentes nas bases. Na figura 34 são apresentadas telas dos sites brasileiro, mexicano e americano com alguns resultados das buscas utilizando palavras-chaves relacionadas à comida. Na tabela 13 são mostrados os tipos de comida mais citados pelos colaboradores.

Tabela 11 – Tipos de comida

Brasil	arroz e feijão, fruta, pão, carne, feijoada, churrasco
México	burritos, chilaquiles, taco, escamoles
EUA	panqueca, hambúrguer, cachorro quente, pizza, sanduíche, torta de abóbora, torta de maçã, sorvete, cheese cake

Uma coisa curiosa é que no banco de dados brasileiro, frutas são citadas como comida muitas vezes, além de ser de muitas variedades. Já nas bases mexicanas e americanas, isso não acontece freqüentemente.

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

Busca Resultados para comer

Conhecimento

Você quase nunca quer um(a) sofá para comer
 Um(a) **bolo** é usado(a) para comer
 Um(a) lanchonete é usado(a) para comer um lanche
 Você quase sempre quer um(a) lousa para comer
 Você geralmente quer um(a) **pastel** para comer
 Você geralmente quer um(a) refeição para comer
 Você geralmente quer um(a) **fruta** para comer
 Um(a) lanchonete é usado(a) para comer
 Um(a) **pão** é usado(a) para comer
 Um(a) **pé de alface** é usado(a) para comer
 Você geralmente quer um(a) **tubarão** para comer
 Você raramente quer um(a) flor para comer
 Um(a) **mesa de frios** é usado(a) para comer
 Você geralmente quer um(a) **pé de alface** para comer
 Um(a) **fruta** é usado(a) para comer e fazer suco

Exemplos de tipos de comidas mencionados por brasileiros: bolo, pastel, fruta, pão, frios

(A)

Search Results for breakfast

Autor	Knowledge
mykel	People eat breakfast when they wake up in the morning
shivel	A jar of jam is used to spread on toast to eat, often for breakfast
captainminus	Make me breakfast
CDavid	People read the paper when they eat breakfast.
southrj1	A carton of eggs is used to cook breakfast
marcus	The picture h0322 gif is of a breakfast
jhe	wolfdances You may want a dozen eggs to be scrambled for breakfast
jhe	rthorson A croissant is used to eat at breakfast
sergio	shroff People rush through breakfast when they are late for work
sergio	celebfinwen A bowl of cereal is used to eat at breakfast
sergio	rguptax In the sentence "You generally find a newspaper in a waiting room" the object 'newspaper' refers to a daily or weekly publication on folded sheets, contains news and articles and advertisements; 'he read his newspaper at breakfast'
sergio	sergio In the sentence "A room where you generally find a newspaper is the study" the object 'newspaper' refers to a
sergio	sergio Al intentar alimentar a un niño, un problema que es posible encontrarse es no le guste la comida
hyz	La tarea que es más urgente de investigar un tema, calentar comida, tomar transporte publico, hacer una presentación y limpiar la mesa es cocinar comida
msgodoi	burritos es una comida
msgodoi	chilaquiles es una comida
msgodoi	taco es una comida
msgodoi	escamoles es una comida
msgodoi	potaje es una comida
msgodoi	pollo es parte de una comida
msgodoi	carne es parte de una comida

Exemplos de tipos de comidas mencionados por americanos: geléia (jam), torradas (toast), ovos (eggs), croissant,

(B)

Exemplos de tipos de comidas mencionados por mexicanos: burritos, chilaquiles, taco, escamoles, frango (pollo), carne

(C)

Figura 34 – Resultado de buscas nos sites do Brasil(A), EUA(B) e México (C)

O que as pessoas comem em cada refeição?

Diferenças entre o que se come em cada refeição também foram notadas. Na tabela 14 é mostrado o que foi considerado senso comum sobre o que se come em cada refeição em cada cultura.

Tabela 12 – O que as pessoas comem em cada refeição?

	Brasil	México	EUA
Café da manhã	Pão, leite, café	tamales, ovos com molho apimentado	panquecas, rosquinhas, ovo e bacon
Almoço	arroz e feijão, carne, salada, ovo, vegetais	frango, carne assada, massa, chilaquiles, barbacoa, tacos	fluffer nutters, hambúrguer, cachorro quente, pizza, sanduíche, wafers
Janta	Arroz e feijão, sopa, salada, sanduíche	tamales e atole, quesadillas, café com biscoito, pão com feijão	Bife e ovos, frango assado, sopa de moluscos, purê de batatas
Sobremesa	Fruta, fruta em calda, sorvete, chocolate	Arroz com leite, churros com chocolate, nozes com mel, doce de coco	torta de abóbora, torta de maçã, sorvete, cheese cake, chocolate, creme

É possível notar que brasileiros preparam uma comida mais leve no café da manhã, comparada com a comida mexicana e a americana servida de manhã. Enquanto no Brasil é encontrado pão, leite e café, no México é comum encontrar ovos com molho apimentado; e ovo e bacon na mesa dos americanos.

Outro fato que se pode notar através dos dados mostrados na tabela é que os mexicanos parecem gostar de comida feita com farinha, pois eles citaram vários pratos feitos com esse ingrediente, tais como, tamales, massa, chilaquiles, tacos, quesadillas, pão e churros.

A respeito de sobremesas, brasileiros associam sorvete com algo gelado e são relutantes em consumir no inverno ao contrário de alguns países mais frios.

Também se pode perceber que os americanos parecem gostar de tortas e outras sobremesas assadas, tais como, torta de abóbora, torta de maçã e cheese cake.

Chocolate para sobremesa foi lembrado pelas três culturas, pois elas aparecem com frequência nas três bases.

Comidas e bebidas para ocasiões especiais

Celebrações de Natal, aniversário e festas foram assuntos que os colaboradores se lembraram também. Na tabela 15 pode se ver os principais tipos de comidas e bebidas citados para essas ocasiões.

Tabela 13 – Comidas e bebidas para ocasiões especiais

	Festa	Natal	Aniversário
Brasil	salgados, doces, bolo, carne, cerveja	peru, lombo, carneiro	Bolo
México	cerveja, tequila	romeritos, bacalhau, macarrão	Bolo
EUA	cerveja, vodka	molho de cranberry, salada de abacaxi, pudim congelado de Natal	Bolo

É interessante notar que no Brasil e no México parecem ser comum as pessoas lembrarem mais dos pratos salgados no natal enquanto nos Estados Unidos, as comidas doces parecem ser mais apreciadas e lembradas. Como se pode ver na tabela 15, todos os pratos citados por brasileiros são salgados: peru, lombo e carneiro. O mesmo ocorre com os mexicanos: romeritos, bacalhau e macarrão. Já os americanos citaram várias comidas doces: molho de cranberry, salada de abacaxi, pudim congelado de Natal.

Um outro fato interessante constatado é que a cerveja é apreciada nas festas pelos colaboradores dos três países analisados.

Também todos parecem gostar de celebrar o aniversário comendo um bolo de aniversário.

Analisando os dados fornecidos pelos colaboradores do projeto Open Mind Common Sense (OMCS) dos três países, foi possível perceber que existem diferenças

nos hábitos alimentares nesses países estudados. Com o intuito de investigar se a diversidade cultural era encontrada não somente entre países diferentes, mas também em grupos sociais distintos, foram feitas algumas análises dos fatos armazenados na base brasileira considerando as possíveis culturas existentes de acordo com os perfis dos colaboradores.

A tabela 16 mostra alguns fatos recuperados da base brasileira considerando um possível grupo cultural: pessoas de idades diferentes.

Tabela 14 – Comidas mencionadas pelos brasileiros

Entre 13 a 17 anos	alimento, comida, peixe, arroz
Mais de 60 anos	carne, salada, lanches, doces, pudim, fruta, garrafa de vinho tinto seco, terrina com sopa, jarra de suco de uva, travessa com batatas fritas, travessa com carne assada, garrafa de vinho tinto espumante, travessa com frango ao molho pardo, travessa com canapés, travessa com peixe assado, jarra de água, travessa com ovos fritos, terrina com sopa, travessa com ostras, travessa com carne com batatas, travessa com bifés a milanesa, jarra com suco de frutas

Foi possível verificar que as pessoas entre 13 a 17 anos ao inserir fatos relacionados a jantar, elas citam coisas simples, como comida, peixe, arroz. Já as pessoas com mais de 60 anos citam alimentos mais elaborados, como travessa de ostras, garrafa de vinho tinto, carne com batatas, dando a impressão de querer descrever suas comidas com mais detalhes.

6.3. Justificativa

Neste trabalho os dados coletados através dos sites OMCS foram utilizados para realizar comparações entre os hábitos alimentares dos brasileiros, americanos e mexicanos. Os dados foram analisados para verificar se os fatos de senso comum, coletados através da Web, de um grupo de pessoas apresentavam algumas características culturais do grupo. As análises iniciais indicam que as bases OMCS

expressam diferenças culturais, como mostradas nas comparações feitas anteriormente. Os dados mostraram interessantes características entre as culturas estudadas, observadas através das análises realizadas, podendo citar alguns exemplos, tais como:

- mexicanos têm suas refeições mais tarde que os demais;
- brasileiros lembram de frutas com mais frequência que os demais;
- e todos gostam de chocolate, cerveja e bolo no aniversário.

Além das diferenças encontradas entre países diferentes, as análises feitas dos fatos coletados de pessoas classificadas por idade, sexo, grau de escolaridade, cidade ou estado de origem, também apontaram diferenças em seus comportamentos.

Essas análises e constatações preliminares citadas acima confirmam o potencial do uso desses fatos de senso comum para fornecer um conhecimento sobre questões culturais às máquinas.

6.4. Considerações Finais

Os resultados obtidos nas análises realizadas mostram que os fatos de senso comum coletados através dos sites OMCS identificam diferenças culturais. Desse modo, pode-se acreditar que esse projeto fornece sua colaboração às aplicações que consideram os aspectos culturais, mencionados anteriormente.

O próximo capítulo apresenta uma análise crítica do trabalho, as dificuldades a serem superadas para o efetivo uso de senso comum considerando as diferenças culturais e os trabalhos futuros para este projeto.

CAPÍTULO 7 - CONCLUSÕES

7.1. Considerações iniciais

Foi explorada aqui a possibilidade de fornecer às máquinas mais inteligência considerando as potencialidades do uso de senso comum e a importância das questões culturais no projeto de sistemas interativos.

Este capítulo apresenta as conclusões do trabalho desenvolvido. Na seção 7.2 é apresentada a síntese dos principais resultados obtidos; na seção 7.3 é feita uma análise crítica; na seção 7.4, as dificuldades a serem superadas serão discutidas; na seção 7.5 lista as publicações realizadas e na seção 7.6 são apresentados os trabalhos futuros.

7.2. Síntese dos Principais Resultados

Este trabalho propôs explorar as possibilidades do uso de conhecimento de senso comum para reconhecer diferenças culturais. Esse recurso ajudará a projetar interfaces mais adequadas, recuperar informações mais relevantes em buscas automatizadas e facilitar a comunicação entre as pessoas .

Comparado com outras áreas de IA, é relativamente pequeno o número de trabalhos usando conhecimento de senso comum considerando aspectos da vida humana (LIU E SIGH, 2004A). Dar esse tipo de raciocínio às máquinas é considerado por eles um dos maiores desafios da área.

Foi mostrado que o senso comum coletado e armazenado nas bases de dados do projeto Open Mind Common Sense (OMCS) reflete aspectos culturais. Essa informação pode ser utilizada para facilitar a interação do usuário com o computador,

pois a máquina pode fazer suposições sobre o que o usuário pode estar querendo fazer numa determinada situação.

Este estudo pode dar uma importante contribuição para pesquisas que buscam a melhoria na comunicação mediada por computadores. Pois este trabalho mostra como constatar tais diferenças e como produzir software que exibem tais diferenças. Acredita-se que fornecendo às pessoas mais informações sobre outras culturas, elas estarão mais aptas a começar e a desenvolver uma conversa com membros de outras culturas, e conseqüentemente, terão impressões mais positivas uma sobre as outras.

7.3. Análise Crítica

A obtenção de senso comum através da Web foi uma idéia inovadora encontrada para adquirir a grande quantidade de informações que se deseja.

Pelo que foi notado, essa abordagem funcionou com sucesso, principalmente no Brasil. Os brasileiros se mostraram bastantes interessados em colaborar com o projeto. Foi impressionante o grande número de colaboradores que se cadastraram e a grande quantidade de fatos inseridos em alguns meses.

Um dos motivos desse sucesso foi a divulgação do trabalho na mídia. Logo depois da divulgação, percebeu-se que muitas pessoas acessaram o site OMCS. Outro fato percebido é que muitos usuários entram uma única vez. Então, um recurso que também foi adotado é o envio de e-mails aos colaboradores, com uma mensagem convidando-os para continuar contribuindo.

A identificação de diferenças culturais é considerada um grande desafio, já que normalmente o conhecimento cultural é implícito e é difícil perceber o que é dependente culturalmente.

Além disso, fornecer às máquinas os diferentes modos de utilizar o conhecimento, para que possam pensar, planejar, explicar, prever e todas as outras habilidades mentais de um ser humano é extremamente complexo.

Nesse contexto, o projeto OMCS no Brasil utiliza as funções para manipulação das redes semânticas com o intuito de identificar tais características culturais implícitas e de dar mais capacidade de raciocínio aos computadores.

Este estudo também mostrou uma preocupação dos profissionais da área da Computação com aspectos humanos, como a utilização dos atributos definidos para classificar culturas, que visam analisar o comportamento dos grupos de pessoas.

O conhecimento sobre características culturais pode ajudar a construir programas mais adequados para cada grupo social, considerando suas particularidades.

7.4. Dificuldades a serem superadas

Considerar aspectos culturais em aplicações computacionais não é uma tarefa fácil, já que as crenças, atitudes e valores de um grupo mudam com o tempo. Por isso, é preciso ter uma inserção contínua de dados na base de conhecimento, porém o trabalho de coletar os dados que revelam as características culturais, e disponibilizar esse conhecimento às aplicações que consideram tais diferenças, pode ser caro e cansativo.

Outro problema encontrado é o desempenho do gerador das redes semânticas. Como o analisador de língua portuguesa, o Curupira, não disponibiliza a forma canônica, foi preciso construir um normalizador. Com isso, dois dicionários da língua portuguesa são carregados todas as vezes que for necessário gerar ou manipular a rede, provocando uma grade lentidão. O desafio é encontrar uma solução para esse problema.

Neste trabalho, o protótipo construído

Apesar da importância dessas questões levantadas em relação a consideração de aspectos culturais nos projetos de interfaces, alguns desenvolvedores ainda encontram uma grande dificuldade de obter apoio para realizar pesquisas a respeito da cultura dos usuários que irão interagir com a interface projetada (MARCUS, 2002).

Outra questão é em relação aos dados da rede semântica. Saber se os dados inseridos por diversos colaboradores da Web geram uma rede semântica confiável capaz de representar todo o conhecimento de senso comum que um humano possui; entender o que é e o que não é senso comum e saber a quantidade de dados necessária para conseguir uma base completa de senso comum são algumas dúvidas que podem surgir.

Apesar de muitos dados pareçam ser inúteis e desprezíveis, acredita-se que elas expressam a opinião e a crença das pessoas. Quanto ao tamanho, enquanto isso não é concretizado, acredita-se que um pouco de senso comum é melhor que nenhum quando se trata de programas que consideram o contexto.

7.5. Publicações

Artigos aprovados:

- Anacleto, J.; Lieberman, H.; Tsutsumi, M.; Neris, V.; Carvalho, A.; Espinosa, J.; Zem-Mascarenhas, S. Can Common Sense uncover cultural differences in computer applications? In WCC 2006 - 19th IFIP World Computer Congress 2006. 20-25 de Agosto de 2006. Santiago- Chile.

Artigos submetidos e em revisão até a data da defesa:

- Knowledge Representation of Cultural Characteristics and Cultural Differences. Submetido em AAAI 2006 - 21th National Conference on

Artificial Intelligence - American Association for Artificial Intelligence.

Notificação: 14 de Abril de 2006.

- Providing Interaction among Culturally Different People Using Common Sense. Submetido em CSCW 2006 - 20th Computer Supported Cooperative Work. Notificação: 02 de Junho de 2006.
- Using Common Sense to Recognize Cultural Differences. Submetido em SBIA 2006 - 18th Brazilian Artificial Intelligence Symposium. Notificação 19 de Maio de 2006.

7.6. Trabalhos futuros

Este trabalho foi realizado considerando o domínio relacionado aos hábitos alimentares com uma base de dados obtidos em 7 meses. Como trabalho futuro, pode-se investigar o senso comum de diferentes culturas utilizando um número maior de fatos e também outros domínios pode ser estudados além do considerado neste estudo.

Com isso, mais detalhes podem ser investigados sobre fatores culturais e como eles afetam a comunicação intercultural. Como mencionado anteriormente, uma comunicação intercultural exige que as pessoas entendam o estilo e a norma de outras culturas (SETLOCK, FUSSELL E NEUWIRTH , 2004). Essas características poderiam ser exibidas explicitamente na interface do programa responsável por promover a interação entre pessoas de culturas diferentes.

Em relação ao uso de senso comum para apoiar ferramentas de *e-learning*, pode-se explorar a possibilidade de utilizar o senso comum de pessoas de uma determinada faixa etária e nível de escolaridade para verificar as preferências que esse grupo cultural possui. Desse modo, pode-se considerar esse conhecimento para (a)

melhorar as buscas feitas pelos alunos em um determinado tema (b) selecionar o tipo de material mais adequado para o perfil do aluno.

Quanto aos efeitos que as cores causam no comportamento das pessoas, o senso comum em relação a esses efeitos poderiam ser verificados e comprovados utilizando experimentos através de projetos de interfaces considerando essa informação.

REFERÊNCIAS

- ANACLETO ET AL., ANACLETO, J.; LIEBERMAN, H.; TSUTSUMI, M.; NERIS, V.;
2006A CARVALHO, A.; ESPINOSA, J.; ZEM-MASCARENHAS, S.
Can Common Sense uncover cultural differences in computer
applications? **Proceedings** WCC 2006. Santiago- Chile.
Disponível em:
<[http://www.media.mit.edu/~lieber/Publications/Cultural-
Differences-IFIP06.pdf](http://www.media.mit.edu/~lieber/Publications/Cultural-Differences-IFIP06.pdf)> . Acesso em Janeiro de 2006.
- ANACLETO ET AL., ANACLETO, J.; TSUTSUMI, M.; LIEBERMAN, H.;
2006B CARVALHO, A.; NERIS, V.; ESPINOSA, J.; ZEM-
MASCARENHAS, S. Providing Interaction among Culturally
Different People Using Common Sense. **Submetido** em CSCW
2006.
- ANACLETO ET AL., ANACLETO, J.; NERIS, V.; CARVALHO, A.; ZEM-
2006C MASCARENHAS, S. TSUTSUMI, M.; LIEBERMAN, H.;
ESPINOSA, J.; Applying Common Sense to Distance Learning:
the Case of Home Care Education. **Submetido** em WWW 2006.
- BAILEY, GURAK, E BAILEY, B.P.; GURAK, L.J.; KONSTAN, J.A. An examination
KONSTAN, 2001 of trust production in computer-mediated exchange. **Proceedings**
Human Factors and the Web 2001 Conference, 2001, US.
Disponível em:
<[http://www.optavia.com/hfweb/7thconferenceproceedings.zip/bai
ley.pdf](http://www.optavia.com/hfweb/7thconferenceproceedings.zip/bailey.pdf)>. Acesso em: Janeiro de. 2006.
- BIRUKOV, BLANZIERI BIRUKOV, A., BLANZIERI, E, GIORGINI, P. An AgentBased
E GIORGINI, 2005 Recommendation System for Web Search. **Proceedings**. AAMAS
2005, Holanda. Disponível em:
<<http://dit.unitn.it/~implicit/papers/source/ic-aamas2005.pdf>>

Acesso em: Janeiro de. 2006.

- BORGAM, 1992 BORGMAN, C. L. Cultural Diversity In Interface Design. ACM SIGCHI Bulletin, 1992, Volume 24, Número 4. Disponível em: <<http://portal.acm.org/citation.cfm?id=142167.142177>>. Acesso em: Janeiro de. 2006.
- BROWN E LEVINSON, 1987 BROWN, P., LEVINSON, S. Universals in Language Usage: Politeness phenomena. In E. Goody (Ed.), Questions and politeness (p. 56-289). **Cambridge University Press** (1987).
- BUCHDID, SILVEIRA, SILVA, 2005 BUCHDID, S.B, SILVEIRA, L.M, SILVA, J.C.A, Análise de Aplicação de Cores em Projeto de EAD: Um Estudo de Caso. **Proceedings** Simpósio nacional de tecnologia e sociedade. 2005.
- CERATTI, 2005 CERATTI, M. Pesquisa - Mente quase aberta. Outubro de 2005. **Correio Brasiliense**. Disponível em: <<http://www.mundoinfo.com.br/informatica/noticias/noticias/principal?codnoticia=05-131007&princ=05-2610-01>>
- CHAU ET AL, 2002 CHAU, P. Y.K., COLE, M., MASSEY, A. P., MONTOYA-WEISS, M., O'KEEFE, R. M. Cultural Differences In The Online Behavior Of Consumers, **Communications Of The Acm**, Vol. 45, No. 10, 2002.
- CLIPPING DE JORNAL, 2005 **Clipping de Jornal**. Ensine um computador a pensar. 23 de outubro de 2005. Disponível em: <<http://www.clipping.ufrgs.br/online-jornal2.php?cod=16575&cliente=5882&datajor=23/08/2005>>
- COMCIÊNCIA, 2005 **COMCIÊNCIA**. UFSCar traz projeto Common Sense do MIT ao Brasil. 04 de Outubro de 2005. Disponível em:

<http://www.comciencia.br/noticias/2005/10/inteligencia_artificial.htm>

- DENISON, 2003 DENISON , D.C. Guess who's smarter. **Boston Globe Online**, 25 Mar. 2003. Disponível em: <<http://web.media.mit.edu/~lieber/Press/Globe-Common-Sense.html>>. Acesso em: Dezembro de 2005
- DICIONÁRIO AURÉLIO, 1995 DICIONÁRIO AURÉLIO BÁSICO DA LÍNGUA PORTUGUESA. **Dicionário**. Editora Nova Fronteira. 1995.
- ESPINOSA, LIEBERMAN, 2005 Espinosa, J. H., Lieberman, H. Using Common Sense Reasoning to Find Cultural Differences in Text. **White paper**. 2005. <http://agents.media.mit.edu/projects/culture/culture_whitepaper.doc>. Acesso em Janeiro de 2006.
- FAABORG LIEBERMAN, 2005 E FAABORG, A.; LIEBERMAN, H. A GOAL-ORIENTED WEB BROWSER. **Proceedings CHI 2006**. Disponível em: <<http://web.media.mit.edu/~lieber/Publications/GoalOrientedWebBrowser.pdf>>. Acesso em Janeiro de 2006.
- HALL, 1976 HALL, E. T. **Beyond Culture**. New York: Anchor Books/Doubleday, 1976.
- HOFSTEDE, 1980 HOFSTEDE, G. **Culture's consequences**. Beverly Hills, CA: Sage, 1980.
- HOFSTEDE, 1991 HOFSTEDE, G. **Cultures and Organizations: Software of the Mind**, London: McGraw-Hill. 1991.
- HOFSTEDE, 1997 HOFSTEDE, G. **Cultures and Organizations: Software of the**

Mind, McGraw-Hill, New York, 1997.

- IDG NOW!, 2005 **IDG NOW**. Projeto ensina fatos comuns ao computador. 22 de Agosto de 2005. Disponível em: <<http://idgnow.uol.com.br/AdPortalv5/ComputacaoPessoalInterna.aspx?GUID=8A387842-076D-47BE-92C0-605F7EDA28A7&ChannelID=2000014>>
- ISBISTER ET AL., 2000 ISBISTER, K; NAKANISHI, H; ISHIDA, T; NASS, C. Helper Agent: Designing an Assistant for Human-Human Interaction in a Virtual Meeting Space. **Proceedings** SIGCHI conference on Human factors in computing systems. 2000. Disponível em: <http://portal.acm.org/ft_gateway.cfm?id=332407&type=pdf&coll=&dl=acm&CFID=15151515&CFTOKEN=6184618>. Acesso em Janeiro de 2006.
- KLUCKHOHN E F. STRODTBECK, 2000 F. KLUCKHOHN AND F. STRODTBECK, **Variations in Value Orientations**. Evanston, IL: Row Peterson, 1961.
- LACERDA, 2005 LACERDA, P. Artificial e cada vez mais inteligente. **Revista Oi**. 22 de outubro de 2005. Disponível em: <<http://www.revistaoi.com.br/nova/noticias.asp?caso=2&id=379&veio=capa&botao=novidades4>>
- LENAT ET AL., 1990 LENAT, D. B.; GUHA, R. V.; PITTMAN, K.; PRATT, D.; SHEPHERD, M. Cyc: toward programs with common sense. **Communications of the ACM**, v. 33, n. 8, 1990. Disponível em: <<http://portal.acm.org>>. Acesso em: Dez. 2005
- LIEBERMAN ET AL., 2004 LIEBERMAN, H.; LIU, H. SINGH, P.; BARRY, B. Beating common sense into interactive applications. **AI Magazine**, v. 25, n. 4, p. 63-76, 2004. Disponível em:

<[http://web.media.mit.edu/~push/ Beating-Common-Sense.pdf](http://web.media.mit.edu/~push/Beating-Common-Sense.pdf)>.

Acesso em: Dez. 2005

- LIEBERMAN E SCHMANDT, 2005 LIEBERMAN H.; SCHMANDT C. Common Sense Reasoning For Multi-Lingual Communication. Massachusetts: MIT/Media Laboratory Software Agents Group, 2005. **Internal Document.**
- LIU E SINGH, 2004A LIU, H.; SINGH P. ConceptNet: A Practical Commonsense Reasoning Toolkit. **BT Technology Journal**, v. 22, n. 4, p. 211-226, 2004. Disponível em: <<http://web.media.mit.edu/~push/ConceptNet-BTTJ.pdf> >. Acesso em: Dez. 2005
- LIU E SINGH, 2004B LIU, H.; SINGH, P. Commonsense Reasoning in and over Natural Language. **Proceedings** INTERNATIONAL CONFERENCE ON KNOWLEDGE-BASED INTELLIGENT INFORMATION & ENGINEERING SYSTEMS (KES'04), 8TH, 2004. Disponível em: <[http://web.media.mit.edu/~push/ CommonsenseInOverNL.pdf](http://web.media.mit.edu/~push/CommonsenseInOverNL.pdf)>. Acesso em: Jan. 2006
- MACGREGOR, HSIEH E KRUCHTEN, 2005 MACGREGOR, E., HSIEH, Y., KRUCHTEN, P. **Proceedings** ACM SIGSOFT Software Engineering Notes: Human and Social Factors of Software Engineering (HSSE), Volume 30 Issue 4. 2005. Disponível em: <<http://portal.acm.org/citation.cfm?id=1082983.1083116&coll=GUIDE&dl=GUIDE&CFID=15255001&CFTOKEN=53732813> >
- MARCUS E GOULD, 2000 MARCUS, A.; GOULD, E. W. Cultural Dimensions and Global Web User-Interface Design. **Proceedings** Interactions, 2000. Disponível em: <<http://portal.acm.org/citation.cfm?id=345238>>. Acesso em Janeiro de 2006.

- MARCUS, 2002 MARCUS, A. Culture Class vs. Culture Clash. **Proceedings Interactions**, v. 9, n. 3. US (New York, NY): ACM Press, Mar. 2002. p. 25-28. Coluna Fast forward. Disponível em: <<http://portal.acm.org/>>. Acesso em: Jan. 2006
- MARTINS, MARTINS, R. T.; HASEGAWA R.; NUNES, M. G. V.
HASEGAWA, NUNES **Curupira: Um Parser Funcional Para A Língua Portuguesa.**
2002 2002
- MCCARTHY, 1959 McCARTHY, J. Programs with Common Sense. **Proceedings The Teddington Conference On Mechanization Of Thought Processes**, 1959. Disponível em: <<http://www-formal.stanford.edu/jmc/mcc59.pdf>>. Acesso em: Nov. 2005.
- MESQUITA AND MESQUITA, B., AND FRIJDA, N.H. (1992) "Cultural Variations
FRIJDA 1992 in Emotions: A Review," **Psycho-logical Bulletin**, Vol. 112, No. 2, pp. 179-204.
- MINSKY, 1986 MINSKY, M. The Society of Mind. Simon and Schuster, New York, 1986.
- MONTYLINGUA, 2005 MontyLingua – A Free, Commonsense-Enriched Natural Language Understander for English. Disponível em: <<http://web.media.mit.edu/~hugo/montylingua/index.html>>. Acesso em: Dezembro de 2005.
- SETLOCK, FUSSELL E SETLOCK, L. D.; FUSSELL, S. R.; NEUWIRTH C. Taking it out
NEUWIRTH, 2004 of context: collaborating within and across cultures in face-to-face settings and via instant messaging. **Proceedings CSCW 2004**. Disponível em: <<http://www.cs.cmu.edu/~sfussell/pubs/Manuscripts/p291->

setlock.pdf >. Acesso em Janeiro de 2006.

- SILVA, 2005A SILVA, M. M. Ensine um computador a pensar. **Jornal Estado de São Paulo**. 22 de agosto de 2005. Disponível em: <http://www.link.estadao.com.br/index.cfm?id_conteudo=4632>
- SILVA, 2005B - SILVA, M. M. Ensine um computador a pensar. **Jornal da Tarde**. 23 de agosto de 2005. Disponível em: <http://www.link.estadao.com.br/index.cfm?id_conteudo=4632>
- SILVEIRA, BUCHDID, SILVA, 2005 SILVA, L. M., BUCHDID, S. B., SILVA, J. C. A., Metodologia de Aplicação de Cores no Projeto WEB. **Proceedings WebMedia 2005**.
- SINGH, 2002A SINGH, P. The OpenMind Commonsense Project. **KurzweilAI.net**, 2002. Disponível em: <<http://web.media.mit.edu/~push/OMCSProject.pdf>>. Acesso em: Dez. 2005
- SINGH, 2002B SINGH, P. The public acquisition of commonsense knowledge. **Proceedings AAAI Spring Symposium On Acquiring (And Using) Linguistic (And World) Knowledge For Information Access**. 2002. Disponível em: <<http://web.media.mit.edu/~push/AAAI2002-Spring.pdf>>. Acesso em: Dez.2005
- SINGH ET AL., 2004 SINGH, P.; BARRY, B.; LIU, H. Teaching machines about Everyday Life. **BT Technology Journal**, v. 22, n. 4, p. 227-240. Disponível em: <<http://web.media.mit.edu/~push/Teaching-Machines-BTTJ.pdf>>. Acesso em: Dez. 2005.
- SPINK (2002) SPINK, A. U.S. Versus European Web Searching Trends. **Proceedings SIGIR Forum- Fall 2002**, Vol. 36, No. 2. Disponível

em: < <http://www.acm.org/sigs/sigir/forum/F2002/spink.pdf> >.
Acesso em Janeiro de 2006.

SUN, 2001

SUN, H. Building A Culturally-Competent Corporate Web Site: An Exploratory Study of Cultural Markers In Multilingual Web Design. **Proceedings** International Conference On Computer Documentation. 2001. Disponível em: < http://portal.acm.org/ft_gateway.cfm?id=501536&type=pdf >.
Acesso em Janeiro de 2006.

ÚLTIMO SEGUNDO,
2005

ÚLTIMO SEGUNDO. UFSCar lança programa que ensina PC a pensar. 22 de outubro de 2005. Disponível em: < http://ultimosegundo.ig.com.br/materias/mundovirtual/2088001-2088500/2088477/2088477_1.xml>

ZORN, 2005

ZORN, I. Do culture and technology interact?: overcoming technological barriers to intercultural communication in virtual communities. **Proceedings** ACM SIGGROUP Bulletin, Volume 25 Issue 2. 2005 Disponível em: < http://portal.acm.org/ft_gateway.cfm?id=1067723&type=pdf >.
Acesso em Janeiro de 2006.

Anexos

ANEXO A

Este anexo tem por objetivo apresentar os campos e códigos utilizados nos dicionários que compõem o Penn Treebank Tagset, utilizado no MontyLingua. A tabela abaixo não apresenta qualquer modificação..

Tag	Description	Example
CC	coordinating conjunction	and
CD	cardinal number	1, third
DT	determiner	the
EX	existential there	<i>there is</i>
FW	foreign word	d'hoevre
IN	preposition/subordinating conjunction	in, of, like
JJ	adjective	green
JJR	adjective, comparative	greener
JJS	adjective, superlative	greenest
LS	list marker	1)
MD	modal	could, will
NN	noun, singular or mass	table
NNS	noun plural	tables
NNP	proper noun, singular	John
NNPS	proper noun, plural	Vikings
PDT	predeterminer	<i>both</i> the boys
POS	possessive ending	friend's
PRP	personal pronoun	I, he, it
PRP\$	possessive pronoun	my, his
RB	adverb	however, usually, naturally, here, good
RBR	adverb, comparative	better
RBS	adverb, superlative	best
RP	particle	give <i>up</i>
TO	to	<i>to go, to him</i>
UH	interjection	uhhuhhuhh
VB	verb, base form	take
VBD	verb, past tense	took
VBG	verb, gerund/present participle	taking
VBN	verb, past participle	taken
VBP	verb, sing. present, non-3d	take
VBZ	verb, 3rd person sing. present	takes
WDT	wh-determiner	which
WP	wh-pronoun	who, what
WP\$	possessive wh-pronoun	whose
WRB	wh-abverb	where, when

ANEXO B

Este anexo tem por objetivo apresentar os campos e códigos utilizados nos dicionários que compõem o Unitex-PB – DELAF-PB, utilizado no Normalizador construído.

Estrutura das entradas:

Palavra, canônica.Classe+traços:flexão

Cada verbete poderá estar classificado em mais de uma classe gramatical, neste caso haverá uma entrada para cada classe.

Substantivo

- Classe: N
- Gênero:
 - m: masculino
 - f: feminino
- Número:
 - s: singular
 - p: plural
- Grau:
 - A: aumentativo
 - D: diminutivo
 - (podendo ser nulo)
- Estrutura:
 - Entrada, canônica.N: grau gênero número
- Exemplos:

menino: menino,menino.N:ms

meninos: meninos,menino.N:mp

meninão: meninão,menino.N:Ams

lápiz: lápis,lápis.N:ms:mp

ajuda: ajuda,ajuda.N:fs

Adjetivo

- Classe: A
- Gênero:
 - m: masculino
 - f: feminino
- Número:
 - s: singular
 - p: plural
- Grau:
 - A: aumentativo
 - D: diminutivo
 - S: superlativo
 - (podendo ser nulo)
- Estrutura:
 - Entrada,canônica.A: grau gênero número
- Exemplos:
 - bonito: bonito,bonito.A:ms
 - bonitas: bonitas,bonito.A:fp
 - aprazível: aprazível,aprazível.A:ms:fs
 - simples: simples,simples.A:ms:mp:fs:fp
 - igual: igual,igual.A:ms:fs

amabilíssimo: amabilíssimo,amável.A:Sms

Artigo

- Classe: DET+Art
- Tipo:
 - Def: Definido
 - Ind: Indefinido
- Gênero:
 - m: masculino
 - f: feminino
- Número:
 - s: singular
 - p: plural
- Estrutura:
 - Entrada,canônica.DET+Art+Tipo:gênero número
- Exemplos:
 - o: o,o.DET+Art+Def:ms
 - umas: umas,um.DET+Art+Ind:fp

Preposição

- Classe: PREP
- Estrutura:
 - Entrada,canônica.PREP
- Exemplos:
 - ante: ante,ante.PREP
 - de: de,de.PREP

Conjunção

- Classe: CONJ

- Estrutura:
Entrada,canônica.CONJ

- Exemplos:
mas: mas,mas.CONJ
mais: mais,mais.CONJ
mal: mal,mal.CONJ

Numeral

- Classe: DET+Num
- Tipo:
C: cardinal
O: ordinal
M: multiplicativo
F: Fracionário
L: Coletivo
- Gênero:
m: masculino
f: feminino
- Número:
s: singular
p: plural
- Estrutura:
Entrada,canônica.DET+Num:tipo gênero número
- Exemplos:
segundo: segundo,segundo.DET+Num:Oms
duplo: duplo,duplo.DET+Num:Mms

Pronome

- Classe: PRO
- Tipo:
 - Dem: Demonstrativo
 - Ind: Indefinido
 - Rel: Relativo
 - Int: Interrogativo
 - Tra: Tratamento
 - Pos: Possessivo
 - Pes: Pessoal
- Forma:
 - A: Forma Acusativa
 - D: Forma Dativa
 - N: Forma Nominativa
 - O: Forma Oblíqua
 - R: Forma Reflexa
 - (podendo ser nulo)
- Pessoa:
 - 1: Primeira Pessoa
 - 2: Segunda Pessoa
 - 3: Terceira Pessoa
- Gênero:
 - m: masculino
 - f: feminino
- Número:
 - s: singular
 - p: plural

- Estrutura:
Entrada, canônica. PRO+Tipo: forma pessoa gênero número
- Exemplos:
senhora: senhora, senhor. PRO+Tra: 3fs
eu: eu, eu. PRO+Pes: N1ms: N1fs

Verbo

- Classe: V
- Tempo:
 - W: Infinitivo
 - G: Gerúndio
 - K: Particípio
 - P: Presente do Indicativo
 - I: Pretérito Imperfeito do Indicativo
 - J: Pretérito Perfeito do Indicativo
 - F: Futuro do Presente do Indicativo
 - Q: Pretérito mais que Perfeito do Indicativo
 - S: Presente do Subjuntivo
 - T: Imperfeito do Subjuntivo
 - U: Futuro do Subjuntivo
 - Y: Imperativo
 - C: Futuro do Pretérito
- Pessoa:
 - 1s: eu
 - 2s: tu
 - 3s: ele
 - 1p: nós

2p: vós

3p: eles

- Estrutura:

Entrada,canônica.V:tempo pessoa

- Exemplos:

cantaríamos: cantaríamos,cantar.V:C1p

cantarias: cantarias,cantar.V:C2s

cantaria: cantaria,cantar.V:C1s:C3s

Advérbio

- Classe: ADV

- Estrutura:

Entrada,canônica.ADV

- Exemplos:

abaixo: abaixo,abaixo.ADV

misericordiosissimamente:

misericordiosissimamente,misericordiosissimamente.ADV

mesmo: mesmo,mesmo.ADV

Prefixos

- Classe: PFX

- Estrutura:

Entrada,canônica.PFX

- Exemplos:

super: super,super.PFX

pós: pós,pós.PFX

sub: sub,sub.PFX

Siglas

- Classe: SIGL
- Estrutura:
Entrada,canônica.SIGL
- Exemplos:
ONU: ONU,ONU.SIGL
PDT: PDT,PDT.SIGL
OTAN: OTAN,OTAN.SIGL
USP: USP,USP.SIGL

Abreviaturas

- Classe: ABREV
- Gênero:
m: masculino
f: feminino
- Número:
s: singular
p: plural
- Estrutura:
Entrada,canônica.ABREV:gênero número
- Exemplos:
ml: ml,ml.ABREV:ms
mm: mm,mm.ABREV:ms

Interjeição

- Classe: INTERJ
- Estrutura:
Entrada,canônica.INTERJ
- Exemplos:

Ah: Ah,Ah.INTERJ

Ih: Ih,Ih.INTERJ

Olá: Olá,Olá.INTERJ

Oi: Oi,Oi.INTERJ