Comparação de Perfís de Usuários Coletados através do Agente de Interface PersonalSearcher*

Gustavo A. Giménez Lugo[1]

Analía Amandi[2]

Jaime Simão Sichman[1]

[1] **Universidade de São Paulo**
Escola Politécnica
Laboratório de Técnicas Inteligentes
Av. Prof. Luciano Gualberto, 158 tv. 3
05508-900 São Paulo- São Paulo, SP, Brasil
e-mail: {gustavo.lugo, jaime.sichman}@poli.usp.br

[2] **Universidad Nacional del Centro de la Prov. de Bs. As.**
Facultad de Ciencias Exactas - ISISTAN Research Institute
C.P. 7000 - Tandil, Buenos Aires, Argentina
e-mail: amandi@exa.unicen.edu.ar

Resumo

Agentes de interface aplicados à recuperação de informação tentam detectar as preferências dos usuários de forma a usar este conhecimento para auxiliá-los em tarefas que envolvam busca de informação. As preferências registradas, tratadas como sendo o perfil de cada usuário, são usadas como guias nos processos de busca. O agente PersonalSearcher [Godoy, 2001, Godoy and Amandi, 2000], desenvolvido no ISISTAN, é um agente de interface deste tipo. Uma linha de pesquisa que objetiva potenciar os resultados obtidos através de agentes deste tipo, em geral isolados, propõe um enfoque cooperativo entre os mesmos. Este enfoque, na forma de sistema multi-agentes (SMA), propiciaria a troca de conhecimentos, permitindo o acesso ao resultado de experiências alheias que potencialmente enriqueceriam os resultados obtidos por agentes individuais. Para possibilitar tal troca de conhecimentos, é necessário determinar a forma pela qual as preferências registradas possam ser comparadas. No presente trabalho, são descritos os resultados experimentais obtidos a partir da implementação de um algoritmo de comparação de diferentes perfís de usuários, considerando a forma na qual este conhecimento é armazenado pelo PersonalSearcher. Como medida de similaridade entre conceitos, foi adotado um modelo retirado da literatura que leva em conta não somente as palavras que descrevem os conceitos, como também a sua vizinhança semântica. Descreve-se também o modelo de similaridade, bem como os ajustes necessários para sua adoção experimental.

Palavras-chave: partilha de conhecimento em sistemas multi-agentes, agentes de interface, recuperação de informação.

*Trabalho desenvolvido no contexto do Projeto de Cooperação Internacional Argentina-Brasil CAPES/SCyT. A estada do primeiro autor na Argentina, no período de setembro a dezembro de 2001, foi financiada por bolsa CAPES, número de processo BEX0510/01-7.

Sumário

Lista de Figuras

Capítulo 1

Introdução

O *PersonalSearcher* [Godoy, 2001, Godoy and Amandi, 2000] é um agente de informação inteligente. O mesmo está projetado para auxiliar o usuário na filtragem de informação durante buscas na Internet. Para tanto, é construída uma representação das preferências do usuário, denominada perfil. Atualmente, estes agentes agem de forma isolada. Uma alternativa para melhorar a sua eficácia na busca é tentar adquirir conhecimento sobre temas correlatos àqueles que aparecem nos seus perfís particulares, agindo como parte de um Sistema Multi Agentes (SMA), pois este conhecimento pode estar disponível em outros agentes. O primeiro passo nesta direção é estabelecer qual a forma de comparar quantitativamente os perfís de usuários na sua forma corrente. É este o objetivo das experiências descritas neste trabalho.

O documento está organizado como descrito a seguir: No restante deste capítulo, são apresentados os conceitos básicos relacionados com agentes de informação. No capítulo 2, é detalhado o modelo denominado MD3 [Rodríguez, 2000] que será utilizado como métrica para medir a similaridade entre conceitos que pertencem a perfís, i.e. ontologias, de usuários diferentes. No capítulo 3, é apresentado o algoritmo implementado para percorrer os perfís a serem comparados, aplicando o cálculo de similaridade a todos os pares de conceitos de ambas as ontologias de forma a mapear aqueles que são candidatos para uma eventual partilha. Na seqüência, o capítulo 4 detalha as condições experimentais de escolha das coleções de teste usadas na geração dos perfís e da sua comparação, assim como os ajustes que foram necessários para adaptar o modelo de similaridade às características do PersonalSeacher, comentando os dados obtidos e discorrendo brevemente sobre algumas linhas de pesquisa que podem ser exploradas a partir dos resultados. Finalmente, o capítulo 5 apresenta as conclusões obtidas dos experimentos realizados.

1.1 Agentes de informação

A despeito da sua difusão, o termo agente não possui, ele próprio uma definição clara e consensual. De todo modo o conceito pode ser visto como uma abstração útil, usada em ciência da computação para referenciar artefatos que possuem algumas características em comum. Agentes são usualmente processos executando de forma contínua, que sabem o que fazer e quando fazê-lo. Agentes comunicam-se com outros agentes, fazendo requisições e executando as tarefas requisitadas. De acordo com [Jennings and Wooldridge, 1998], um agente possui uma longa lista de propriedades, dentre as quais podem ser destacadas as seguintes:

- *Autonomia*: operam sem intervenção humana direta, tendo controle sobre as suas ações;
- *Habilidade social*: comunicam-se através de uma linguagem comum com outros agentes

e inclusive seres humanos;

- *Reatividade*: percebem o seu ambiente e reagem a mudanças nele ocorridas;
- *Proatividade*: são capazes de exibir comportamento dirigido a um objetivo, tomando a iniciativa de atingí-lo.

Devido à enorme quantidade de informação disponível, na Internet e em outros repositórios, e ao escasso tempo de que o usuário em geral dispõe para achar informação relevante, um tipo de agente que tem sido alvo de intensas pesquisas é o chamado Agente de Informação Inteligente([Klusch, 2001], [Levy and Weld, 2000], [Kobayashi and Takeda, 2000], [Mladenic, 1999], [Finin et al., 1998]).

Agentes de informação são definidos por Klusch [Klusch, 2001], como sendo entidades computacionais de software que:

- Podem acessar uma ou múltiplas fontes de informação, distribuídas e heterogêneas;
- Adquirem, mediam e mantêm proativamente informação relevante em representação do usuário ou de outros agentes, preferivelmente numa forma *just-in-time*.

De uma forma geral podem ser:

- *Cooperativos*: quando os agentes colaboram na execução de tarefas;
- *Não-cooperativos*: se os agentes não colaboram para executar tarefas.

Ambos os tipos de agentes mencionados, por sua vez, podem ser:

- *Racionais*: se o agente pode decidir racionalmente quando e como executar tarefas, comprar e negociar para aumentar o seu benefício;
- *Adaptativos*: quando o agente é capaz de adaptação a um meio em mudança (usuário, rede, informação);
- *Móveis*: se o agente pode transportar-se de forma autônoma pela Internet para executar tarefas em servidores diferentes.

Em [Mladenic, 1999] são focalizados sistemas que empregam agentes de informação que usam técnicas de Aprendizado de Máquina (AM) ou Mineração de Dados (MD) para fornecer assistência ao usuário na busca de informação na web ou representando-o em tarefas mais simples. Dois tipos de métodos usados com freqüência no desenvolvimento de Sistemas Multi-Agentes (SMA) aplicados à recuperação de informação baseados em AM são [Mladenic, 1999], [Finin et al., 1998]:

- *Enfoques Baseados em Conteúdo*: quando os agentes procuram ítens similares àqueles preferidos pelo usuário baseados na comparação do conteúdo. As suas raízes estão no domínio de Recuperação de Informação (RI), sendo populares quando o alvo do sistema desenvolvido está constituído por dados textuais como documentos web ou *news*. Podem ser aplicados com sucesso a usuários isolados;
- *Enfoques Colaborativos*: também chamados de *aprendizado social* [Maes, 1994], parte de suposição de que existe um conjunto de usuários usando o sistema. Ao invés de computar similaridade entre ítens, o sistema computa a similaridade entre os usuários, recomendando ítens que usuários similares acharam interessantes.

O desempenho deste tipo de sistemas reside crucialmente na modelagem que fazem dos seus usuários.

1.2 Perfís de usuários

Cada agente, seja ele parte de um SMA ou não, pode ser considerado como possuindo uma visão particular do ambiente. Esta visão pode ser explicitada através de uma *ontologia*, definida em [Guarino, 1997] como correspondendo a uma teoría lógica projetada para dar conta do significado pretendido para certo vocabulário. [Guarino and Giaretta, 1995] sugerem que uma conceituação é:

"an intensional semantic structure which encodes the implicit rules constraining the structure of a piece of reality"

Uma ontologia é então o comprometimento com uma conceituação particular do mundo. Esta definição refina a de [Gruber, 1995] que estabelece que uma ontologia é uma especificação explícita de uma conceituação .

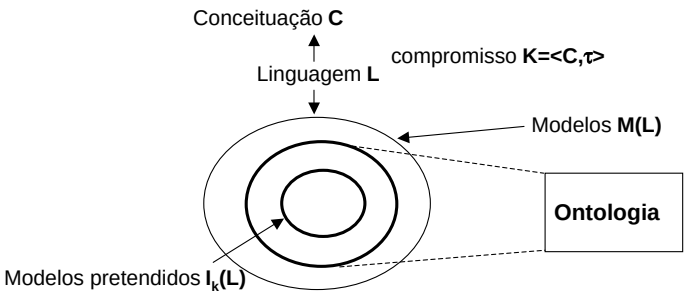

Figura 1.1: Relacionando ontologia, linguagem e conceituação [Guarino, 1997].

O uso de ontologias para a explicação de conhecimentos implícitos é um enfoque viável para superar o problema de heterogeneidade semântica. A interoperabilidade entre agentes pode ser atingida reconciliando as diferentes visões de mundo através do compromisso com uma ontologia comum [Wache et al., 2001,]. Na mesma linha de raciocínio, ontologias podem ser usadas como ferramentas de mediação entre as requisições de agentes e as fontes de informação externas ao sistema e/ou para a filtragem de informação relevante. Agentes de informação podem usar ontologias para representar explicitamente perfís de usuários isolados [Pretschner, 1999, Chaffee and Gauch, 2000].

1.3 PersonalSearcher

O agente de interface PersonalSearcher [Godoy and Amandi, 2000, Godoy, 2001] é um agente de interface dedicado à recuperação de informação da *web*. Utiliza a técnica de Raciocínio Baseado em Casos para elaborar um perfil das preferências temáticas do usuário. Este perfil é incremental e de geração completamente automática, sendo usado para expandir as consultas do usuário com termos relevantes, assim como para filtrar as páginas mais relevantes do conjunto obtido, através de consultas às máquinas de busca de propósito geral tais como Google, Yahoo ou Altavista.

Perfís de usuários são gerados pelo PersonalSearcher a partir da observação do comportamento de busca do usuário. Dados como o tempo de leitura de documentos, endereço de origem, conteúdo, etc. são usados para descrever o documento como um *caso* o qual é comparado a outros casos previamente armazenados. A partir de um certo grau de semelhança entre estes casos, os mesmos são agregados em um mesmo agrupamento. Os agrupamentos são monitorados e no caso de atingirem certas condições de invariância, um grupo de palavras representativas do agrupamento é escolhido para gerar um *tema*, que a partir daí será usado

para recortar o espaço de opções na hora de verificar se um documento recuperado da web é relevante para o usuário.

Os temas sucessivamente gerados pelo PersonalSearcher estão organizados numa *hierarquia*. As hierarquias temáticas obtidas pelo PersonalSeacher podem ser consideradas como representando ontologias particulares. Nesta linha, os temas passam a ser tratados como sendo conceitos que são descritos por meio de atributos, no caso as palavras que descrevem um tema.

Capítulo 2

Medida de similaridade

Na literatura, existem alguns trabalhos que tratam da comparação de ontologias [Mitra et al., 2000] [Fridman and Musen, 1999][McGuinness et al., 2000] e que referenciam ferramentas específicas. Estas ferramentas permitem a construção de ontologias complexas, sendo projetadas para interagir com especialistas humanos. Em contraste com esses enfoques mais sofisticados que permitem resultados mais precisos, em [Rodríguez, 2000] é apresentado um modelo, relativamente simples, denominado MD3, para comparar ontologias. Além de permitir a automatização completa do processo de comparação, combina três formas de medição e pode ser aplicado com poucas modificações às hierarquias presentes nos perfís de usuários do PersonalSearcher ou a ontologias bem mais detalhadas. Embora o uso do modelo MD3 possa acarretar perdas na precisão do resultado obtido, no caso de ser considerada uma sociedade de agentes de informação é essencial que o processo de comparação possa ser efetivado com o maior grau de automatização possível, caso contrário o usuário poderia ser incessantemente interpelado para decidir conflitos que o distrairiam de seu verdadeiro objetivo.

2.1 O modelo utilizado

O modelo MD3 avalia a similaridade entre classes pertencentes a hierarquias de conceitos diferentes. Utilizou-se este modelo para estabelecer a similaridade entre os conceitos que se encontram em perfís de usuários diferentes de PersonalSearcher. Na seqüência, o modelo é apresentado com maior detalhamento.

Este modelo leva em conta que duas hierarquias de conceitos independentes (ontologias) são consideradas como estando conectadas através de uma classe (imaginária) mais geral.

O citado modelo propõe medir a similaridade entre conceitos combinando *feature-matching* (*matching* levando em conta atributos ou características) e distância semântica.

O valor de similaridade global $S(a, b)$ entre dois conceitos a e b, pertencentes a ontologias diferentes, digamos p e q, é dado pela equação:

$$S(a^p, b^q) = w_l * S_l(a^p, b^q) + w_u * S_u(a^p, b^p) + w_n * S_n(a^p, b^q) \tag{2.1}$$

onde S_l, S_u e S_n são respectivamente as similaridades léxica (de nomes), de atributos e de vizinhança semântica entre os conceitos a e b, e w_l, w_u e w_n são os pesos referentes a cada uma destas componentes para calcular o valor da similaridade global.

Os valores de w_l, w_u e w_n dependem das características das ontologias a serem comparadas. Somente aspectos de especificação comuns a ambas as ontologias podem ser usados para verificar o grau de similaridade. Assim, se numa das ontologias o nome usado para identificar um conceito (digamos "ABC0027") não carrega informação a respeito do conceito,

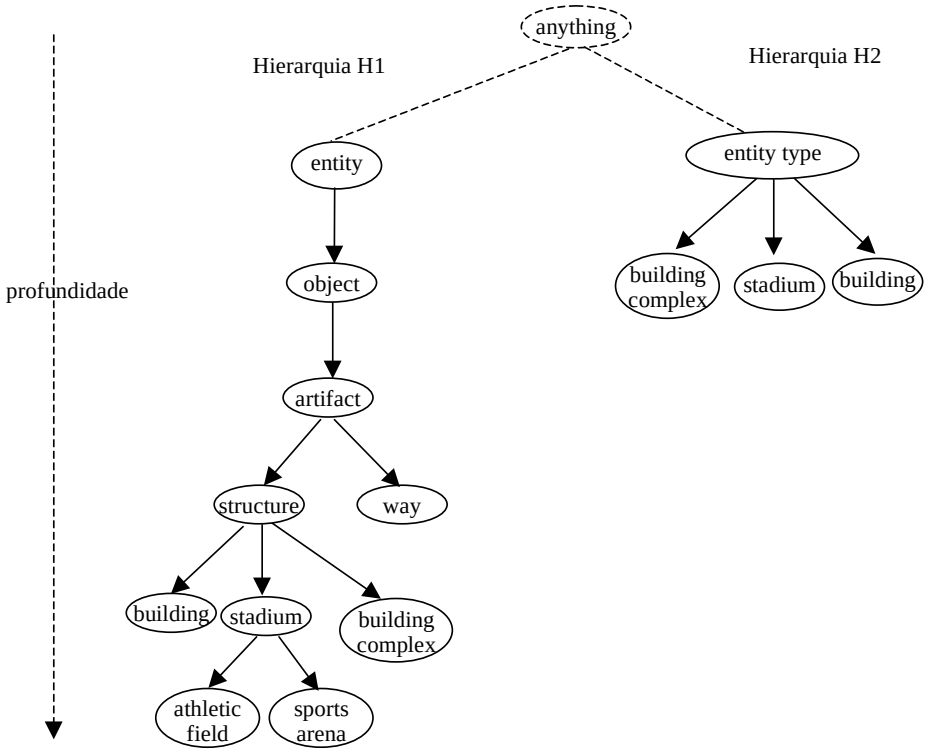

Figura 2.1: Conexão de duas ontologias usando uma raiz comum imaginária, adaptado de [Rodríguez, 2000]

sendo apenas um identificador, não faz sentido usar este identificador como parâmetro na hora de comparar este conceito com um conceito pertencente a outra ontologia onde o nome em sí mesmo ajuda a descrever o conceito (digamos "laranja"). A soma dos pesos deve ser igual a um.

A similaridade léxica referente aos nomes, embora possa apontar similaridade entre duas classes, pode ser influenciada por palavras que apresentam polisemia. Este é o motivo pelo qual deve ser complementado com a similaridade baseada em vizinhança semântica e em similaridade de atributos.

Nas funções de similaridade $S_i(a^p, b^q)$, a e b são classes (no caso do PersonalSearcher, temas) e i denota o tipo de similaridade abordado (de nome ou léxico, de atributos e de vizinhança semântica).

Sejam A e B os conjuntos de atributos de a^p e de b^q. O processo de *matching* determina a cardinalidade ($|\ |$) da interseção ($A \cap B$) e da diferença ($A - B$) entre A e B.

$$S_i(a^p, b^q) = \frac{|A \cap B|}{|A \cap B| + \alpha(a^p, b^q) * |A - B| + (1 - \alpha(a^p, b^q)) * |B - A|} \tag{2.2}$$

O valor de α é calculado em função da profundidade das classes (a classe a pertencendo à ontologia p e a classe b pertencendo à ontologia q):

$$\alpha(a^p, b^q) = \begin{cases} \frac{prof(a^p)}{prof(a^p)+prof(b^q)} & \text{Se } prof(a^p) \leq prof(b^q) \\ 1 - \frac{prof(a^p)}{prof(a^p)+prof(b^q)} & \text{Se } prof(a^p) > prof(b^q) \end{cases} \tag{2.3}$$

Um aspecto que deve ser levado em consideração é que a similaridade $S(a^p, b^q)$ entre conceitos não é uma relação necessariamente simétrica ([Tversky, 1977] apud [Rodríguez, 2000]), e.g. a expressão "um *hospital* é similar a um *prédio*" é aceita de forma mais geral que a expressão "um *prédio* é similar a um *hospital*". Tem sido sugerido que a distância percebida de um conceito mais geral (neste caso *prédio*) para outro menos geral (no caso *hospital*) é maior que a distância percebida do conceito menos geral (*hospital*) ao conceito mais geral (*prédio*) ([Krumhansl, 1978] apud [Rodríguez, 2000]).

2.2 Cálculo de similaridade léxica

Considerem-se as ontologias da figura 2.1. O cálculo da similaridade entre os conceitos *building* da hierarquia $H1$ e *building_complex* da hierarquia $H2$ é descrito a seguir.

Sejam:

$a = building$

$b = building\_complex$

$A = \{\text{building}\}$

$B = \{\text{building\_complex}\}$

$| A \cap B | = | \{\text{building}\} | = 1$

$| A \text{ - } B | = | \{ \} | = 0$

$| B \text{ - } A | = | \{\text{complex}\} | = 1$

Como $prof(building^{H1}) > prof(building\_complex^{H2})$, pois $5 > 2$, tem-se que:

$$\alpha(a^p, b^q) = 1 - \frac{prof(building^{H1})}{prof(building^{H1}) + prof(building\_complex^{H2})} = 1 - \frac{5}{5+2} = 0.28 \quad (2.4)$$

e portanto:

$$1 - \alpha(a^p, b^q) = 0.72 \quad (2.5)$$

Assim:

$$S_l(building^{H1}, building\_complex^{H2}) = \frac{|\{building\}|}{|\{building\}| + 0.28 * |\{\}| + 0.72 * |\{complex\}|} \quad (2.6)$$

$$S_l(building^{H1}, building\_complex^{H2}) = \frac{1}{1 + 0 + 0.72} = 0.58 \quad (2.7)$$

Para casos onde são usados conjuntos de sinônimos (Sys) para a e b, o cálculo é feito levando em conta as palavras com maior similaridade. Por exemplo, se fossem considerados:

$Sys(building) = \{building, edifice\}$

$Sys(building\_complex) = \{building\ complex\}$

O valor calculado de 0.58 para a similaridade léxica entre *building* e *building_complex* prevalecerá sobre a similaridade entre *edifice* e *building_complex* (cujo valor é zero).

Este mesmo cálculo de similaridade considerando os conceitos $stadium^{H1}$ e $stadium^{H2}$ nas hierarquias teria como resultado 1.0, independente do valor de α.

2.3 Cálculo de similaridade de atributos

Para este cálculo, serão consideradas representações sintáticas das características relacionadas a um conceito. Assim, no caso do PersonalSearcher, elas vêm a ser as palavras que descrevem um tema. Será ignorado o caso de similaridade de termos compostos (e.g. entre *lane* e *number of lanes*).

2.3.1 Cálculo considerando apenas palavras associadas a um tema

Considerem-se novamente as ontologias da figura 2.1. O cálculo da similaridade de atributos dos conceitos $stadium^{H1}$ e $stadium^{H2}$ é descrito a seguir.

Sejam:

Atributos($stadium^{H1}$) = { foundation, midfield, playing_field, plate, sports_arena, stands, standing_room, structural_elements, tiered_seats}

Atributos($stadium^{H2}$) = {dressing_room, foundation, midfield, playing_field, spectator_stands, ticket_office}

$a = stadium^{H1}$

$b = stadium^{H2}$

A = {foundation, midfield, playing_field, plate, sports_arena, stands, standing_room, structural_elements, tiered_seats}

B = {dressing_room, foundation, midfield, playing_field, spectator_stands, ticket_office}

$| A \cap B | = |\{foundation, midfield, playing\_field\}| = 3$

$| A - B | = |\{plate, sports\_arena, stands, standing\_room, structural\_elements, tiered\_seats\}| = 6$

$| B - A | = |\{dressing\_room, spectator\_stands, ticket\_office\}| = 3$

Como $prof(stadium^{H1}) > prof(stadium^{H2})$, pois $5 > 2$, tem-se que:

$$\alpha(a^p, b^q) = 1 - \frac{prof(stadium^{H1})}{prof(stadium^{H1}) + prof(stadium^{H2})} = 1 - \frac{5}{5+2} = 0.28 \tag{2.8}$$

e portanto:

$$1 - \alpha(a^p, b^q) = 0.72 \tag{2.9}$$

Assim:

$$S_u(stadium^{H1}, stadium^{H2}) = \frac{3}{3 + 0.28 * 6 + 0.72 * 3} = \frac{3}{3 + 1.68 + 2.16} = 0.44 \tag{2.10}$$

2.3.2 Cálculo considerando sinônimos das palavras associadas a um tema

Caso se considerem sinônimos, o resultado anterior seria ligeiramente alterado, como mostrado a seguir.

Sejam:

Atributos($stadium^{H1}$) = { foundation, midfield, playing_field, plate, sports_arena, stands, standing_room, structural_elements, tiered_seats }

Atributos($stadium^{H2}$) = { dressing_room, foundation, midfield, playing_field, spectator_stands, ticket_office }

Os sinônimos (Sys) de cada expressão, nas respectivas hierarquias a que pertencem são:

$Sys(foundation^{H1}) = \{foundation\}$

$Sys(midfield^{H1}) = \{midfield\}$

$Sys(playing\_field^{H1}) = \{playing\_field, athletic\_field, field\}$

$Sys(plate^{H1}) = \{plate\}$

$Sys(sports\_arena^{H1}) = \{sports\_arena, field\_house\}$

$Sys(stands^{H1}) = \{stands\}$

$Sys(standing\_room^{H1}) = \{standing\_room\}$

$Sys(structural\_elements^{H1}) = \{structural\_elements\}$

$Sys(tiered\_seats^{H1}) = \{tiered\_seats\}$

$Sys(dressing\_room^{H2}) = \{dressing\_room\}$

$Sys(foundation^{H2}) = \{foundation\}$

$Sys(midfield^{H2}) = \{midfield\}$

$Sys(playing\_field^{H2}) = \{playing\_field, athletic\_field, sports\_field\}$

$Sys(spectator\_stands^{H2}) = \{spectator\_stands, stands\}$

$Sys(ticket\_office^{H2}) = \{ticket\_office, box\_office, ticket\_booth\}$

Neste caso, dois atributos são considerados iguais se a interseção dos conjuntos de sinônimos não é vazia. Desta forma, os seguintes atributos são considerados iguais no exemplo:

- $foundation^{H1}$ e $foundation^{H2}$, pois têm como interseção dos seus conjuntos de sinônimos a palavra $foundation$;
- $midfield^{H1}$ e $midfield^{H2}$, pois têm como interseção dos seus conjuntos de sinônimos a palavra $midfield$;
- $playing\_field^{H1}$ e $playing\_field^{H2}$, pois têm como interseção dos seus conjuntos de sinônimos as palavras $playing\_field$ e $athletic\_field$;
- $stands^{H1}$ e $spectator\_stands^{H2}$, pois têm como interseção dos seus conjuntos de sinônimos a palavra $stands$.

Isso altera os seguintes valores:

$|A \cap B| = |\{foundation, midfield, playing\_field, stands\}| = 4$

$|A-B| = |\{plate, sports\_arena, standing\_room, structural\_elements, tiered\_seats\}| = 5$

$|B - A| = |\{dressing\_room, ticket\_office\}| = 2$

Como no caso anterior:

$$\alpha(a^p, b^q) = 0.28 \tag{2.11}$$

Onde

$a = stadium^{H1}$

$b = stadium^{H2}$

e:

$$1 - \alpha(a^p, b^q) = 0.72 \tag{2.12}$$

Assim:

$$S_u(stadium^{H1}, stadium^{H2}) = \frac{4}{4 + 0.28 * 5 + 0.72 * 2} = \frac{4}{4 + 1.40 + 1.44} = 0.59 \tag{2.13}$$

2.4 Cálculo da similaridade de vizinhança semântica

A vizinhança semântica de uma classe, numa rede semântica, é o conjunto de classes cuja distância à classe dada é menor ou igual a um valor especificado. Este valor é chamado de *raio* da vizinhança semântica. A distância entre duas classes é medida como sendo o caminho mais curto, formado pelo menor número de arcos não-dirigidos que conectam as classes. A distância, definida desta forma, é uma métrica que satisfaz a propriedade de minimalidade (a distância de uma classe a sí propria é zero), sendo que a vizinhança semântica de uma classe contém à propria classe. Formalmente:

$$N(a^o, r) = \{c_i^o\} \tag{2.14}$$

Nesta equação a^o e c_i^o são classes de uma ontologia o, r é o raio e $N(a^o, r)$ é a vizinhança semântica de a^o. Tem-se que $\forall c_i^o \in N(a^o, r)$, $d(a^o, c_i^o) \leq r$.

Considerando a figura 2.2, a vizinhança de $stadium^{H1}$ é:

$$N(stadium^{H1}, 1) = \{structure, stadium, athletic\_field, sports\_arena\} \tag{2.15}$$

A noção de similaridade abordada é baseada em *matching* raso (*shallow*, com o mesmo sentido usado para a igualdade rasa em orientação a objetos), associada à avaliação da vizinhança imediata de cada classe, i.e., raio 1. Esta noção difere da noção de *matching* profundo (*deep*, com o mesmo sentido usado para a igualdade profunda em orientação a objetos) que considere vizinhanças semânticas com raio maior que um, onde o *matching* seria baseado na similaridade das folhas das vizinhanças consideradas. Como abuso de notação, nas seções posteriores $S_n(a^p, b^q)$ deve ser lido como $S_n(a^p, b^q, 1)$.

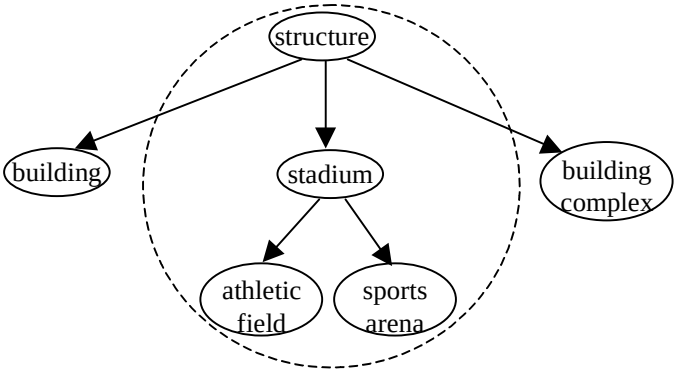

Figura 2.2: Exemplo de vizinhança semântica de raio 1 para a classe *stadium* [Rodríguez, 2000].

Sejam dois conceitos a^p e b^q pertencentes a duas ontologias distintas p e q, onde a vizinhança semântica de raio r de a^p contém n conceitos e a vizinhança semântica de raio r de b^q contém m conceitos e a interseção entre as duas vizinhanças semânticas é denotada por $a^p \cap_n b^q$. O cálculo da similaridade da vizinhança semântica, considerando as respectivas vizinhanças semânticas de raio r, obedece à seguinte fórmula:

$$S_n(a^p, b^q, r) = \frac{|a^p \cap_n b^q|}{|a^p \cap_n b^q| + \alpha(a^p, b^q) * \delta(a^p, a^p \cap_n b^q, r) + (1 - \alpha(a^p, b^q)) * \delta(b^q, a^p \cap_n b^q, r)} \tag{2.16}$$

Onde:

$$\delta(a^p, a^p \cap_n b^q, r) = \begin{cases} |N(a^p, r)| - |a^p \cap_n b^q| & \text{Se } |N(a^p, r)| > |a^p \cap_n b^q| \\ 0 & \text{Em outro caso} \end{cases} \tag{2.17}$$

A interseção entre as vizinhanças semânticas é aproximada pela similaridade de classes nessas vizinhanças:

$$a^p \cap_n b^q = \left[\sum_{i \leq n} \left(max_{j \leq m} S(a_i^p, b_j^q) \right) \right] - \varphi * S(a^p, b^q) \tag{2.18}$$

Com

$$\varphi = \begin{cases} 1 & \text{Se } S(a^p, b^q) = max_{j \leq m}\ S(a^p, b_j^q) \\ 0 & \text{Em outro caso} \end{cases} \tag{2.19}$$

Sendo $S(a^p, b^q)$ uma função assimétrica, a interseção aproximada $a^p \cap_n b^q$ também o é. A interseção aproximada efetua o *matching* de classes com máxima similaridade. Como observado no cálculo de φ, este *matching* exclui a similaridade entre as duas classes que estão sendo efetivamente comparadas (i.e., a^p e b^q) pois esta avaliação seria redundante. Como está expressado na equação 2.19, se a *similaridade da classe* a^p *com a classe* b^q *propriamente* é igual à *similaridade máxima de* a^p *com alguma classe da vizinhança semântica de* b^q, φ

valerá 1 pois certamente deverá ser debitado o valor da similaridade entre a^p e b^q. O valor da similaridade entre duas classes a_i^p e b_j^q, pertencentes às vizinhanças semânticas de a^p e b^q respectivamente, é calculado usando similaridade de léxico e de atributos, segundo a fórmula:

$$S(a_i^p, b_j^q) = w_l * S_l(a_i^p, b_j^q) + w_u * S_u(a_i^p, b_j^q) \tag{2.20}$$

Na fórmula 2.20, os valores de w_l e w_u são considerados inicialmente como sendo iguais a 0.50.

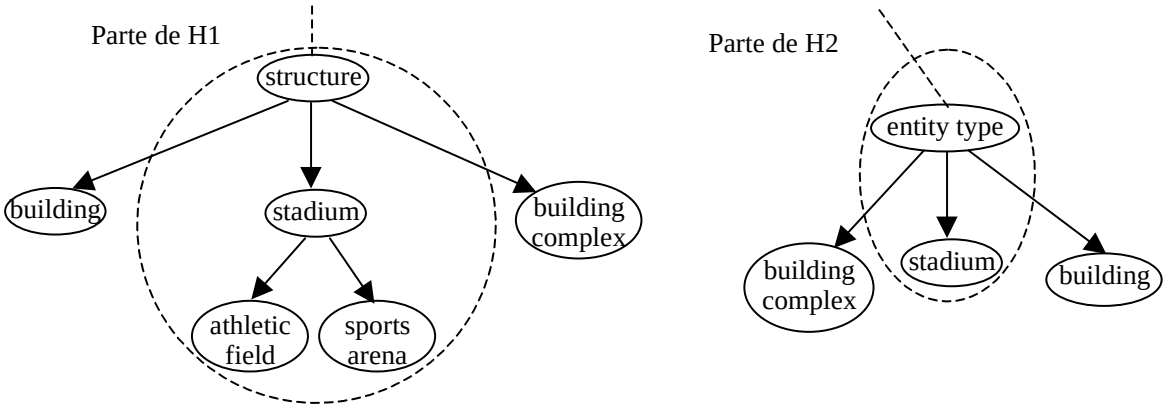

Figura 2.3: Exemplo de vizinhança semântica de raio 1 para as classes $stadium^{H1}$ e $stadium^{H2}$, adaptado de [Rodríguez, 2000].

A título de exemplo, sejam as ontologias apresentadas na figura 2.3. O valor da similaridade de vizinhança semântica é calculado a seguir.

Sejam:

$a = stadium^{H1}$

$b = stadium^{H2}$

$N(stadium^{H1}, 1) =$ { structure, stadium, athletic_field, sports_arena }

$N(stadium^{H2}, 1) =$ { entity_type, stadium }

n (número de classes da vizinhança semântica de $a = stadium^{H1}$) $= 4$

m (número de classes da vizinhança semântica de $b = stadium^{H2}$) $= 2$

- Cálculo de $\left[\sum_{i \leq n}\left(max_{j \leq m} S(a_i^p, b_j^q)\right)\right]$ para a equação 2.18:

  Neste cálculo, os valores de similaridade entre as classes a_i^p e b_j^q consideradas são computados segundo a fórmula 2.20.

  Para $i = 1$ (i.e., considerando a primeira das 4 classes na vizinhaça de $stadium^{H1}$)

  para j=1: $S(structure^{H1}, entity\_type^{H2}) = 0.5 * 0.0 + 0.5 * 0.0 = 0.0$
  para j=2: $S(structure^{H1}, stadium^{H2}) = 0.0$
  $max_{j \leq 2} = 0.0$; $\sum_{1 \leq 4}\left(max_{j \leq m} S(a_i^p, b_j^q)\right) = 0.0$

Para $i = 2$ (i.e., considerando a segunda das 4 classes na vizinhaça de $stadium^{H1}$)

para j=1: $S(stadium^{H1}, entity\_type^{H2}) = 0.0$

para j=2: $S(stadium^{H1}, stadium^{H2}) = 1.0$

$max_{j \leq 2} = 1.0; \sum_{2 \leq 4} \left(max_{j \leq m} S(a_i^p, b_j^q) \right) = 1.0$

Para $i = 3$ (i.e., considerando a terceira das 4 classes na vizinhaça de $stadium^{H1}$)

para j=1: $S(athletic\_field^{H1}, entity\_type^{H2}) = 0.0$

para j=2: $S(athletic\_field^{H1}, stadium^{H2}) = 0.0$

$max_{j \leq 2} = 0.0; \sum_{3 \leq 4} \left(max_{j \leq m} S(a_i^p, b_j^q) \right) = 1.0$

Para $i = 4$ (i.e., considerando a quarta das 4 classes na vizinhaça de $stadium^{H1}$)

para j=1: $S(sports\_arena^{H1}, entity\_type^{H2}) = 0.0$

para j=2: $S(sports\_arena^{H1}, stadium^{H2}) = 0.0$

$max_{j \leq 2} = 0.0; \sum_{4 \leq 4} \left(max_{j \leq m} S(a_i^p, b_j^q) \right) = 1.0$

- Cálculo de $\varphi * S(a^p, b^q)$ para a equação 2.18:

  Para $j = 1$ (i.e., considerando a primeira das 2 classes na vizinhaça de $stadium^{H2}$)

  $S(a^p, b_1^q) = S(stadium^{H1}, entity\_type^{H2}) = 0.0$

  Para $j = 2$ (i.e., considerando a segunda das 2 classes na vizinhaça de $stadium^{H2}$)

  $S(a^p, b_2^q) = S(stadium^{H1}, stadium^{H2}) = 1.0$

  $max_{j \leq 2} S(a^p, b_j^q) = 1.0$

  Desta forma, $S(stadium^{H1}, stadium^{H2}) = 1.0$

  De 2.19, como $S(stadium^{H1}, stadium^{H2}) = max_{j \leq 2} S(a^p, b_j^q)$

  Então $\varphi = 1$

- Cálculo de $a^p \cap_n b^q$, segundo a equação 2.18. Este valor será utilizado na eq. 2.16:

  $a^p \cap_n b^q = \left[\sum_{i \leq n} \left(max_{j \leq m} S(a_i^p, b_j^q) \right) \right] - \varphi * S(a^p, b^q)$

  $a^p \cap_n b^q = 1.0 - 1.0 * 1.0 = 0.0$

- Cálculo de $\delta(a^p, a^p \cap_n b^q, 1)$ para a equação 2.16:

  $|N(a^p, 1)| = 4$

  Como $|N(a^p, 1)| > a^p \cap_n b^q$

  $\delta(a^p, a^p \cap_n b^q, r) = |N(a^p, r)| - a^p \cap_n b^q$

  $\delta(a^p, a^p \cap_n b^q, r) = 4 - 0 = 4$

- Cálculo de $\delta(b^q, a^p \cap_n b^q, 1)$ para a equação 2.16:

  $|N(b^q, 1)| = 2$

Como $|N(b^q, 1)| > a^p \cap_n b^q$

$\delta(b^q, a^p \cap_n b^q, r) = |N(b^q, r)| - a^p \cap_n b^q$

$\delta(b^q, a^p \cap_n b^q, r) = 2 - 0 = 2$

- Cálculo de $\alpha(a^p, b^q)$ para a equação 2.16:

  Como $prof(stadium^{H1}) > prof(stadium^{H2})$, pois $5 > 2$,

  então

  $\alpha(a^p, b^q) = 1 - \frac{prof(stadium^{H1})}{prof(stadium^{H1}) + prof(stadium^{H2})} = 1 - \frac{5}{5+2} = 0.28$

- Cálculo de $S_n(a^p, b^q, 1)$, segundo a equação 2.16:

  $S_n(a^p, b^q, 1) = \frac{|a^p \cap_n b^q|}{|a^p \cap_n b^q| + \alpha(a^p, b^q) * \delta(a^p, a^p \cap_n b^q, r) + (1 - \alpha(a^p, b^q)) * \delta(b^q, a^p \cap_n b^q, r)}$

  Então

  $S_n(stadium^{H1}, stadium^{H2}, 1) = \frac{0}{0 + 0.28 * 4 + 0.72 * 2} = 0$

2.5 Cálculo da similaridade total

Partindo da equação inicial (2.1) e substituindo na mesma os valores obtidos (particularmente escolhendo como valor da similaridade de atributos aquele obtido sem considerar sinônimos), considerando ainda pesos equivalentes para os valores de similaridade ($w_l = w_u = w_n = 0.33$), tem-se:

$S(a^p, b^q) = w_l * S_l(a^p, b^q) + w_u * S_u(a^p, b^p) + w_n * S_n(a^p, b^q)$

$S(stadium^{H1}, stadium^{H2}) = w_l * S_l(stadium^{H1}, stadium^{H2}) + w_u * S_u(stadium^{H1}, stadium^{H2}) + w_n * S_n(stadium^{H1}, stadium^{H2})$

$S(stadium^{H1}, stadium^{H2}) = 0.33 * 1.0 + 0.33 * 0.44 + 0.33 * 0.0$

$S(stadium^{H1}, stadium^{H2}) = 0.48$

Como explicado na seção 2.4, será usado o valor de *raio* igual a 1 na avaliação da vizinhança semântica para o cálculo do respectivo valor de similaridade.

Capítulo 3

Algoritmo de comparação de perfís de usuários

Usando como métrica de similaridade entre conceitos o modelo MD3, apresentado no capítulo 2, foi implementado um algoritmo que efetiva a comparação entre dois perfís.

O algoritmo consta de dois procedimentos, denominados *Percorrer_1* e *Percorrer_2*, sendo ambos essencialmente encaminhamentos em largura em ambas as árvores de temas.

Procedimento 1 Percorrer_1(perfil_1, perfil_2)

```
fila_1.inserir(perfil_1.raiz)
enquanto fila_1 não vazia faça
  no_1 ← fila_1.inicio
  para k = 1 até num_filhos(no_1) faça
    filho_k ← proximo_filho(no_1)
    fila_1.inserir(filho_k)
    fila_vizinhanca.inicializar()
    fila_vizinhanca.inserir(no_1)
    fila_vizinhanca.inserir(filho_k)
    para j = 1 até num_filhos(filho_k) faça
      fila_vizinhanca.inserir(proximo_filho(filho_k))
    fim para
    Percorrer_2(filho_k, fila_vizinhanca, perfil_2)
  fim para
  fila_1.retirar(no_1)
fim enquanto
```

O procedimento *Percorrer_1* percorre em largura o primeiro perfil ($perfil\_1$), considerado como uma referência. Para tanto vai guardando numa fila ($fila\_1$) primeiramente o nó raíz e, na seqüência, todos os filhos da raíz. Para cada filho ($filho\_k$) agregado à fila, é construída uma outra fila ($fila\_vizinhanca$) contendo todos os nós na sua vizinhança semântica de raio 1 (i.e., para um nó *filho_k*, o nó **pai_k** que é o nó pai do nó $filho\_k$, o próprio nó **filho_k** e todos os nós **filhos de filho_k**). Neste ponto o nó $filho\_k$, a $fila\_vizinhanca$ e a àrvore $perfil\_2$ são enviadas a um outro procedimento, *Percorrer_2*, que está encarregado de comparar a similaridade de $filho\_k$ com os conceitos de $perfil\_2$. Após terem sido percorridos todos os nós filhos do nó raíz, o nó raíz é retirado da $fila\_1$. Seguidamente são processados todos os filhos do primeiro filho da raíz, depois os do segundo e assim sucessivamente até ter sido percorrida integralmente toda a árvore $perfil\_1$.

O procedimento *Percorrer_2* recebe um nó do $perfil\_1$ e o compara com todos os nós

Procedimento 2 Percorrer_2(no, fila_vizinhanca, perfil_2)

```
fila_2.inserir(perfil_2.raiz)
enquanto fila_2 não vazia faça
  no_2 ← fila_2.inicio
  para m = 1 até num_filhos(no_2) faça
    filho_m ← proximo_filho(no_2)
    fila_2.inserir(filho_m)
    fila_vizinhanca_2.inicializar()
    fila_vizinhanca_2.inserir(no_2)
    fila_vizinhanca_2.inserir(filho_m)
    para n = 1 até num_filhos(filho_m) faça
      fila_vizinhanca_2.inserir(proximo_filho(filho_m))
    fim para
    elemento_lista.similaridade = S(no, fila_vizinhanca, filho_m, fila_vizinhanca_2)
    no.lista_similaridade.inserir(elemento_lista)
  fim para
  fila_2.retirar(no_2)
fim enquanto
```

do $perfil\_2$. Esta comparação é feita percorrendo a árvore de temas $perfil\_2$ em largura. Inicialmente, é guardada a raíz de $perfil\_2$ na lista que guiará o percurso ($fila\_2$), passando a serem agregados nessa lista sucessivamente todos os filhos ($filho\_m$) da raíz. Para cada um destes filhos é montada uma lista contendo a sua vizinhança semântica de raio 1 ($fila\_vizinhanca\_2$, incluindo o nó **pai** de $filho\_m$, o próprio nó **filho_m** e todos os **filhos do nó filho_m**).

A vizinhança semântica de $filho\_m$, assim como o proprio nó $filho\_m$ são enviados junto com o nó recebido como parâmetro (no, pertencente a $perfil\_1$) e a correspondente vizinhança semântica de no ($fila\_vizinhanca$) para a função $S(no, fila\_vizinhanca, filho\_m, fila\_vizinhanca\_2)$ que efetivamente calcula o valor da similaridade entre o nó no (de $perfil\_1$) e o correspondente nó $filho\_m$ (de $perfil\_2$).

O cálculo do valor de similaridade entre os nós segue as linhas ditadas pelo algoritmo MD3 mostrado no capítulo 2. O modelo MD3 será apenas ajustado segundo as particularidades de um perfil gerado pelo PersonalSearcher:

- Como num perfil de usuário de PersonalSearcher o nome do tema não carrega informação (e.g. "$SBJ234$"), sendo apenas um identificador gerado de forma automática, o peso da similaridade lexical será zero (vide seção 2.1). Passam a ser consideradas apenas as similaridades de atributos e de vizinhança semântica da equação 2.1:

$$S(a^p, b^q) = 0.0 * S_l(a^p, b^q) + 0.5 * S_u(a^p, b^p) + 0.5 * S_n(a^p, b^q) \tag{3.1}$$

  Na versão original as três formas de comparação têm pesos iguais (*0.33*). Ao ser eliminada a similaridade de nomes, os pesos da similaridade de atributos e de vizinhança semântica foram escolhidos como sendo *0.5* no contexto do presente trabalho pois não existiam referências anteriores sobre a influência relativa de cada forma de similaridade considerando os perfís gerados pelo PersonalSearcher. Este é um aspecto que deverá ser explorado em trabalhos futuros, dependendo de uma maior disponibilidade de perfís de usuários de PersonalSearcher tanto em número de perfís quanto na variedade de tópicos neles incluidos.

- Os atributos de um tema em um perfil, i.e. as palavras associadas ao tema, possuem pesos associados que serão levados em consideração. Tais pesos afetam o valor da cardinalidade tanto da interseção ($|A \cap B|$) quanto das diferenças ($|A - B|$ e $|B - A|$) dos conjuntos de atributos dos temas considerados na equação 2.2 quando a mesma é aplicada ao cálculo da similaridade de atributos:

$$S_u(a^p, b^q) = \frac{|A \cap B|}{|A \cap B| + \alpha(a^p, b^q) * |A - B| + (1 - \alpha(a^p, b^q)) * |B - A|}$$

Na equação 2.2 a cardinalidade é inteira. Ao considerar os pesos das palavras a cardinalidade passa a ter como valor um número real. A modo de ilustração, é apresentado o cálculo dos valores da interseção e das diferenças para o mesmo exemplo da seção 2.3 para o cálculo da similaridade de atributos, sem considerar sinônimos.

Sejam:

Atributos($stadium^{H1}$) = { foundation, midfield, playing_field, plate, sports_arena, stands, standing_room, structural_elements, tiered_seats}

Atributos($stadium^{H2}$) = {dressing_room, foundation, midfield, playing_field, spectator_stands, ticket_office}

$a = stadium^{H1}$

$b = stadium^{H2}$

Nos respectivos conjuntos de atributos, os mesmos passam a ser representados por pares ordenados indicando o peso de cada atributo para o conceito correspondente.

A = { (0.7,foundation), (0.8,midfield), (0.9,playing_field), (0.8,plate), (0.7,sports_arena), (0.6,stands), (0.5,standing_room), (0.5,structural_elements), (0.5,tiered_seats) }

B = { (0.5,dressing_room), (0.6,foundation), (0.6,midfield), (1.0,playing_field), (0.6,spectator_stands), (0.3,ticket_office) }

$|A \cap B| = |\{(x, foundation), (y, midfield), (z, playing\_field)\}|$ =x + y + z = $(0.7_A + 0.6_B)/2 + (0.8_A + 0.6_B)/2 + (0.9_A + 1.0_B)/2 = 0.65 + 0.70 + 0.95 = 2.30$

$|A - B|$ = | { (0.8,plate), (0.7,sports_arena), (0.6,stands), (0.5,standing_room), (0.5,structural_elements), (0.5,tiered_seats) } | $= 0.8_A + 0.7_A + 0.6_A + 0.5_A + 0.5_A + 0.5_A = 3.6$

$|B - A|$ = | { (0.5,dressing_room), (0.6,spectator_stands), (0.3,ticket_office)}| $= 0.5_B + 0.6_B + 0.3_B = 1.4$

Capítulo 4

Experimentos

O modelo descrito no capítulo 2, implementado no algoritmo visto no capítulo 3, foi aplicado a alguns perfís de usuários, obtidos fornecendo como entradas ao PersonalSearcher subconjuntos de coleções de páginas web. Nas seções que seguem, são descritas as características das coleções e os subconjuntos usados, os perfís obtidos, assim como a comparação entre estes perfís.

4.1 Características das coleções usadas para a obtenção dos perfís experimentais

Foram usadas duas coleções de páginas web[1] para montar perfís de usuários fictícios. Como o PersonalSearcher mantém uma base de casos para cada perfil, que é atualizada como conseqüência das inferências do agente sobre as ações observadas do usuário (e.g. leitura de uma página web), as páginas pertencentes às coleções foram apresentadas ao PersonalSearcher como se fossem decorrentes da interação de usuários que as consideraram relevantes. Desta forma, o agente passou a considerá-las para análise e categorização em agrupamentos de casos e/ou temas, agregando-as ao banco de casos do usuário fictício correspondente.

A opção de montar perfís a partir de coleções de teste foi escolhida pela necessidade de se ter um controle sobre a forma de geração dos mesmos, assegurando a repetibilidade da experiência. Este fato é especialmente importante quando se tem varios parâmetros que podem ser ajustados no próprio PersonalSearcher, e que influenciam tanto no agrupamento das páginas quanto na geração de temas [Godoy, 2001]:

- Limiar mínimo para agregar um caso (página web) a um agrupamento de casos;
- Número mínimo de novos casos, agregados a um agrupamento com invariância dos atributos (palavras chaves), indicando a mudança de estado de agrupamento para tema;
- Limiar mínimo para classificar um caso sob um tema;
- Limiar mínimo para escolher uma palavra como atributo de um tema.

Ainda, caso os perfís fossem gerados com a intervenção de usuários humanos, o tempo de construção dos perfís necessários seria considerável, sendo requeridas semanas de uso intensivo do PersonalSearcher para cada perfil construído com uma combinação de parâmetros específicos.

As duas coleções de teste escolhidas foram disponibilizadas na Internet pelo grupo de pesquisa em classificação automática de texto da Carnegie Mellon University que as têm usado nas suas próprias experiências. Elas apresentam as seguintes características:

[1] Obtidas no site http://www-2.cs.cmu.edu/afs/cs/project/theo-4/text-learning/www/datasets.html

- A primeira coleção, denominada *WebKB* (originalmente denominada *The 4 Universities Data Set*), consta de 8282 páginas coletadas em janeiro de 1997 dos Departamentos de Ciência da Computação de 4 Universidades dos EUA.

  O grupo de pesquisa citado classificou manualmente estas páginas. Elas estão distribuídas em 7 diretórios principais, que por sua vez contêm 5 subdiretórios cada. O conteúdo trata de cursos, departamentos, projetos, docentes, alunos, etc. pertencentes aos citados departamentos. Os 7 diretórios principais são:

  student
  faculty
  staff
  department
  course
  project
  other

  Cada um dos diretórios principais contém páginas distribuídas em 5 subdiretórios:

  Cornell
  Texas
  Washington
  Wisconsin
  misc

- A segunda coleção, doravante citada apenas com o nome *Sector* (originalmente *Industry Sector Data Set*), consta de 9548 páginas distribuídas em 105 diretórios de forma muito homogênea. Esta coleção apresenta diretórios que contêm, como norma, um número de páginas que varia entre 90 e 100. Cada um destes diretórios corresponde a um setor de atividade industrial (e.g. "*audio.and.video.equipment.industry*").

Em testes do PersonalSearcher anteriores ao presente trabalho, o número de páginas usadas não ultrapassou 250. Neste experimento, se decidiu considerar em cada coleção um subconjunto que contivesse pelo menos o dobro deste número. Os subconjuntos efetivamente utilizados como entrada para o PersonalSearcher em ambas as coleções foram as seguintes:

WebKB: Nenhum dos sete diretórios principais apresentaram características que os diferenciariam, de forma vantajosa ou não, dos outros. Decidiu-se então escolher um de forma aleatória, recaindo a escolha no diretório *Course*, com 973 páginas, ocupando 5.02 Mb, distribuídas nos subdiretórios:

cornell
wisconsin
washington
texas
misc

Sector: Dada a homogeneidade da distribuição das páginas nesta coleção, considerou-se escolher 5 diretórios que permitiriam atingir o número mínimo de páginas proposto. A escolha foi feita então levando em conta a ordem alfabetica dos diretórios. Assim foram consideradas 521 páginas, ocupando um total de 2.84Mb, correspondentes aos seguintes diretórios:

alcoholic.beverages.industry

apparel.accessories.industry

appliance.and.tool.industry

audio.and.video.equipment.industry

auto.and.truck.manufacturers.industry

Nas páginas que se seguem, as referências que aparecem em relação a ambas as coleções devem ser entendidas como tratando apenas dos subconjuntos aqui citados.

4.2 Perfís obtidos

A partir dessas duas coleções, foram gerados três perfís experimentais.

O PersonalSearcher associa para cada tema gerado num perfil um determinado valor, que está relacionado ao tempo de leitura dispendido pelo usuário nas páginas (casos) que estão classificadas sob esse tema em relação ao tempo total de leitura do usuário. Este valor é interpretado pelo agente como sendo o grau de preferência do usuário em relação ao tema. Sendo fictícios os perfís elaborados, foi necessário atribuir aleatoriamente os graus de preferência a cada tema gerado, que equivaleriam à soma dos tempos de leitura das páginas correspondentes.

Os valores escolhidos servem para estabelecer uma ordem no ranking de preferências de temas do usuário, fazendo sentido apenas quando comparados às crenças de usuários reais. Assim, no contexto da experiência realizada, os valores numéricos propriamente não são relevantes e sim o fato de poder levá-los em consideração no modelo, como será visto na seção 4.3.3.

Os perfís obtidos foram os seguintes:

- **UserONE**: foi usada como entrada para o PersonalSearcher a coleção *Sector*.

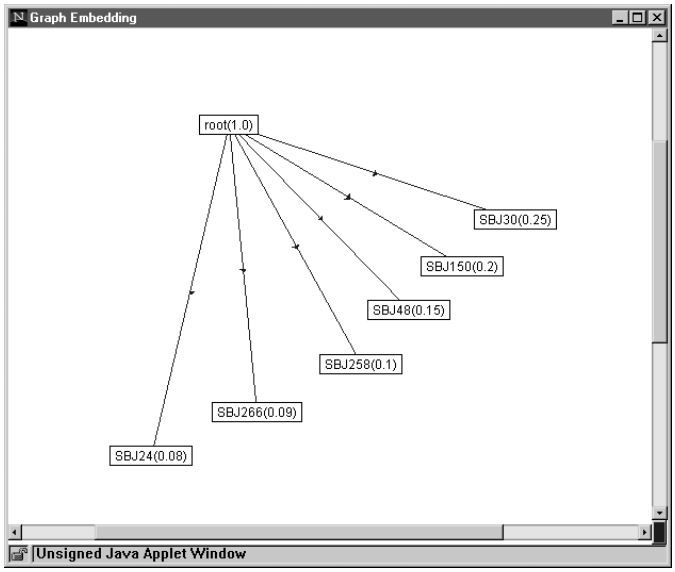

Figura 4.1: Perfil do usuário fictício *UserONE* gerado a partir da coleção *Sector*. Os valores entre parênteses indicam o grau de preferência de cada tema.

Tema	*Atributo (peso do atributo)*
SBJ30	network (1.0) right (1.0) evergreen (1.0) media (1.0) creat (1.0) reserv (1.0)
SBJ150	sorri (1.0) dust (1.0) our (1.0) pardon (1.0) check (1.0)
SBJ48	docum (1.0) found (1.0) accordingli (1.0) hotlist (1.0) locat (1.0) pleas (1.0) move (1.0) ha (1.0)
SBJ266	volvo (1.0)
SBJ258	fiat (1.0)
SBJ24	profil (1.0) innov (1.0) corpor (1.0) entertain (1.0) fall (1.0) lodgenet (1.0) sioux (1.0) employ (1.0)

Tabela 4.1: Atributos dos temas pertecentes ao perfil do usuário UserONE

- **UserTWO**: foi usada como entrada para o PersonalSearcher a coleção $WebKB$.

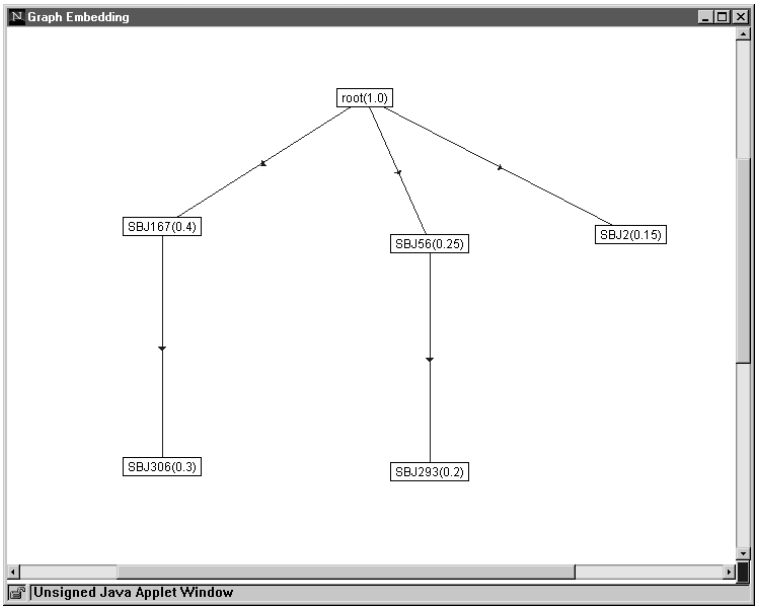

Figura 4.2: Perfil do usuário fictício $UserTWO$ gerado a partir da coleção $WebKB$.Os valores entre parênteses indicam o grau de preferência de cada tema.

Tema	*Atributo (peso do atributo)*
SBJ167	comput (1.0) cp (1.0)
SBJ306	homework (1.0) syllabu (1.0) comput (1.0)
SBJ293	bestavro (1.0) docum (1.0) professor (1.0) prepar (1.0) been (1.0) azer (1.0) comput (1.0) syllabu (1.0) ha (1.0)
SBJ2	washington (1.0) cse (1.0)
SBJ56	move (1.0) docum (1.0) ha (1.0) perman (1.0)

Tabela 4.2: Atributos dos temas pertecentes ao perfil do usuário UserTWO

- **UserTHREE**: foram usadas como entrada para o PersonalSearcher ambas as coleções $Sector$ e $WebKB$, nesta ordem.

  A ordem de entrada foi relevante pois pode-se perceber que os códigos gerados a partir da coleção $Sector$ (i.e. aqueles com identificadores até o $SBJ258$) coincidem com os pertencentes ao perfil do usuário $UserONE$ tanto no identificador como nas palavras que servem de atributos.

  Os códigos que foram decorrentes do processamento da coleção $WebKB$ (aqueles com índices superiores a $SBJ258$) não coincidem com os temas do perfil do usuário $UserTWO$, não apenas pela óbvia diferença nos identificadores, mas também pelas palavras que aparecem como atributos. Isto se explica pela presença de agrupamentos de páginas (casos) decorrentes do processamento da coleção $Sector$ que passaram a gerar temas ao serem adicionadas algumas páginas da segunda coleção. Assim, embora parte da estrutura da nova hierarquia certamente lembra aquela pertencente ao perfil do usuário $UserTWO$, esta aparece "contaminada" por palavras decorrentes do processamento da primeira coleção.

Tema	*Atributo (peso do atributo)*
SBJ614	azer (1.0) syllabu (1.0) bestavro (1.0) comput (1.0)
SBJ48	docum (1.0) found (1.0) accordingli (1.0) hotlist (1.0) locat (1.0) pleas (1.0) move (1.0) ha (1.0)
SBJ857	assign (1.0) syllabu (1.0)
SBJ266	volvo (1.0)
SBJ379	move (1.0) docum (1.0) ha (1.0) perman (1.0)
SBJ150	sorri (1.0) dust (1.0) our (1.0) pardon (1.0) check (1.0)
SBJ30	network (1.0) right (1.0) evergreen (1.0) media (1.0) creat (1.0) reserv (1.0)
SBJ258	fiat (1.0)
SBJ24	profil (1.0) innov (1.0) corpor (1.0) entertain (1.0) fall (1.0) lodgenet (1.0) sioux (1.0) employ (1.0)
SBJ323	washington (1.0) cse (1.0)
SBJ340	syllabu (1.0)
SBJ490	comput (1.0) cp (1.0)

Tabela 4.3: Atributos dos temas pertecentes ao perfil do usuário UserTHREE

Os perfís experimentais foram gerados com os seguintes valores de limiar usados pelo PersonalSearcher:

- **Limiar mínimo para agregar um caso a um agrupamento:** 0.50, usando a fórmula de distância do cosseno para representações vetoriais de documentos [Godoy, 2001]. Originalmente este parâmetro tinha o valor 0.60;
- **Limiar mínimo para classificar um caso sob um tema:** 0.60, usando a fórmula de distância do cosseno considerando apenas as palavras que servem de atributo ao tema [Godoy, 2001]. Originalmente este parâmetro tinha o valor 0.80;

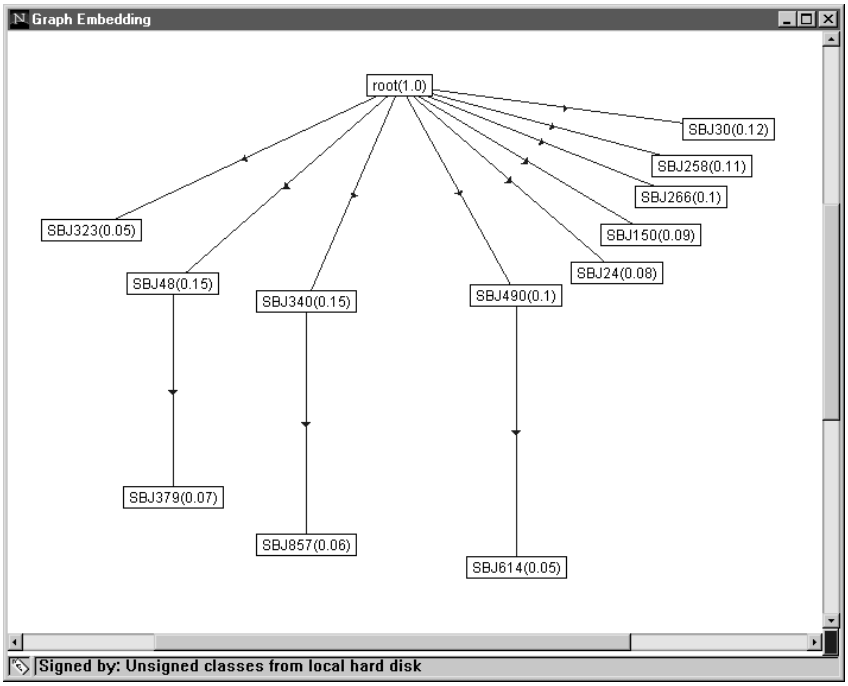

Figura 4.3: Perfil do usuário fictício $UserTHREE$ gerado a partir das coleções $Sector$ e $WebKB$. Os valores entre parênteses indicam o grau de preferência de cada tema.

- **Limiar mínimo para escolher uma palavra como atributo de um tema:** 0.60, o que implica aparecer em 60% dos documentos que estão classificados num agrupamento [Godoy, 2001]. Originalmente este parâmetro tinha o valor 0.80;
- **Número mínimo de novos casos agregados a um agrupamento com invariância dos atributos:** 3, i.e. um novo tema é gerado quando as palavras que servem de atributo a um tema permanecem invariáveis após terem sido agregadas 3 páginas consecutivamente ao mesmo agrupamento de casos [Godoy, 2001]. Originalmente este parâmetro tinha o valor 5.

Inicialmente foram usados os valores originais dos parâmetros do PersonalSearcher, porém os resultados obtidos com ambas as coleções de teste não foram satisfatórios, pois os valores mostraram-se muito limitantes, os agrupamentos de páginas similares não geravam temas e quando o faziam, o número de páginas analisadas era excessivamente alto, na ordem de várias dezenas de páginas, o que certamente acarretaria frustração num usuário. Experimentalmente, os valores foram paulatinamente reduzidos até atingir os resultados mostrados através dos três perfís gerados.

Os perfís confirmam a intuição decorrente da inspeção do conteúdo das coleções usadas. A coleção $Sector$ está distribuída em apenas um nível de diretórios e efetivamente o perfil por ele gerado ($UserONE$) não apresenta uma hierarquia de fato. Já a coleção $WebKB$, por focalizar com maiores detalhes um certo domínio (refletindo isto em diretórios com mais de um nível) gera claramente uma hierarquia de temas (o perfil do usuário $UserTWO$). A combinação de ambas as coleções (seguindo uma ordem estrita entre elas) gerou um perfil (aquele do usuário $UserTHREE$) que mistura as características dos perfís anteriores.

4.3 Comparações dos perfís obtidos

A comparação experimental seguiu três fases. Inicialmente, foram comparados os perfís gerados consigo próprios, de forma a observar o comportamento do algoritmo nesses casos referenciais. Posteriormente, após um ajuste na fórmula de similaridade, foram comparados os três perfís entre si. Finalmente, foram considerados na comparação os valores associados ao grau de preferência de um tema por parte do usuário.

Em todos os casos, as figuras representam telas geradas pelo algoritmo de comparação implementado.

4.3.1 Casos de referência: comparação dos perfís gerados consigo próprios

Para testar inicialmente a corretude do algoritmo, foram escolhidos como casos de base as situações onde a comparação de um perfil é realizada consigo próprio.

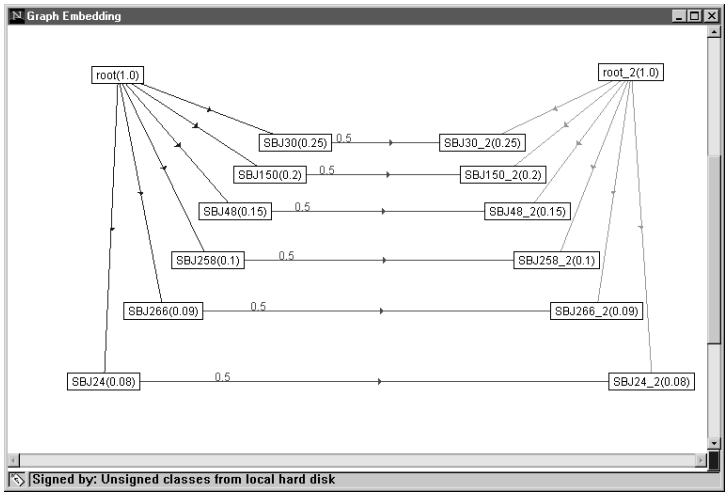

Figura 4.4: Comparação do perfil do usuário $UserONE$ consigo próprio.

As comparações das figuras 4.4, 4.5 e 4.6 mostram detalhes aparentemente contraditórios pois o grau de similaridade dos temas consigo próprios não chega a 1.0, sendo em alguns casos tão baixo quanto 0.50. Isto se deve a temas filhos da raíz que não possuem filhos. Tais temas têm na sua vizinhança semântica apenas o próprio tema, sendo zero o valor da similaridade de vizinhança semântica (e.g. vide $SBJ30$ na figura 4.4). Na medida em que uma vizinhança semântica abrange outros temas além do tema alvo da comparação, os valores de similaridade obtidos são mais elevados (e.g., vide o valor de similaridade de 0.75 atingido entre $SBJ293$ e $SBJ293\_2$ na figura 4.5).

Os temas que são filhos da raíz e que não possuem filhos[2] passaram a ser denominados *temas_isolados*. Os *temas_isolados* são gerados quando um usuário leu um número de páginas sobre um mesmo tópico (e.g. *Esportes*) em número suficiente para gerar um tema, porém este número de leituras não foi suficiente para permitir a especialização em sub-temas (e.g. *Futebol*, *Boxe*, *Vôlei*). A estratégia adotada para levar em conta esta particularidade foi a de identificar os *temas_isolados* e calcular para eles somente a similaridade de atributos quando comparados com outros temas do mesmo tipo, i.e. atribuir peso *1.0* à similaridade de atributos e peso *0.0* à similaridade de vizinhança semântica. Isto equivale a comparar, se e somente se *ambos* os conceitos sendo comparados são *temas_isolados*, apenas as palavras que são atributos desses temas. Se for comparado um tema comum com um *tema_isolado*

[2] E.g. *SBJ30*, *SBJ150*, etc. na figura 4.4; *SBJ2* na figura 4.5

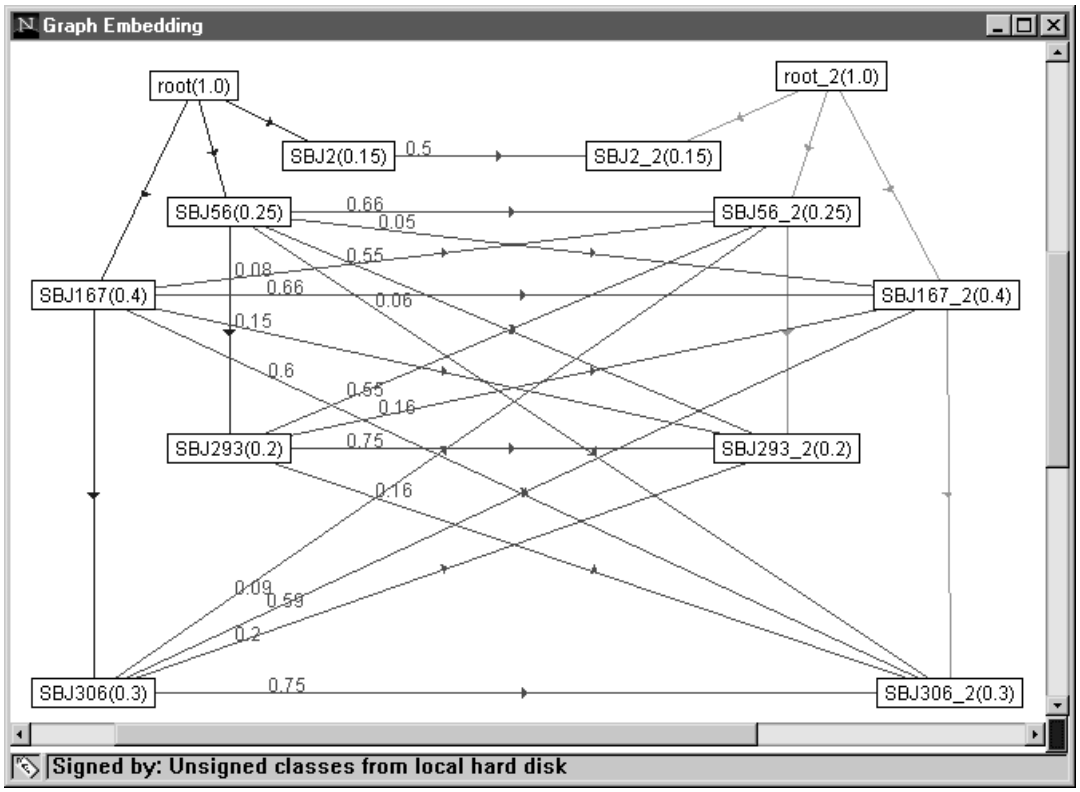

Figura 4.5: Comparação do perfil do usuário $UserTWO$ consigo próprio.

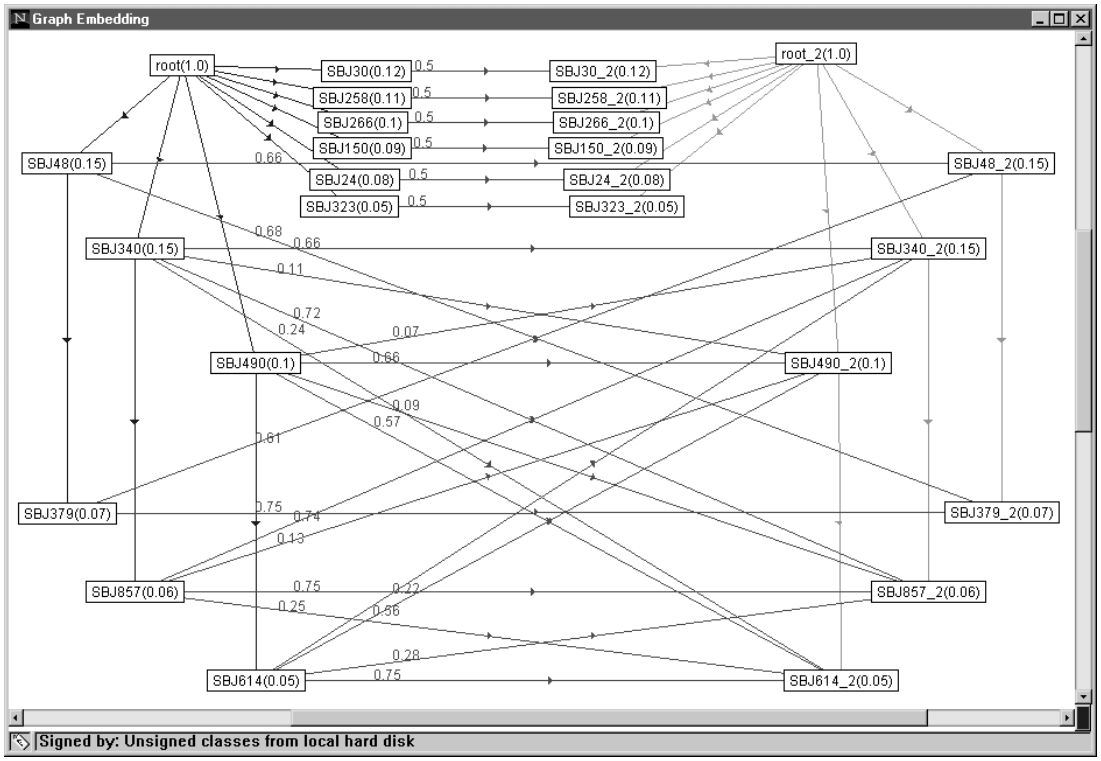

Figura 4.6: Comparação do perfil do usuário $UserTHREE$ consigo próprio.

ou vice-versa, os pesos para a similaridade de atributos e para a similaridade de vizinhança semântica permanecem iguais a 0.5, não alterando a equação 3.1.

4.3.2 Casos alvo: comparação de perfís diferentes

De posse dos resultados de referência, foram confrontados os três perfís experimentais.

A observação das figuras 4.7 e 4.8 confirma a previsão sobre o fato da similaridade não ser simétrica (vide seção 2.4). No caso, a similaridade de $SBJ48$ com $SBJ56\_2$ é 0.25 (figura 4.7), já a similaridade de $SBJ56$ com $SBJ48\_2$ (figura 4.8) é de 0.29.

Ainda, nas figuras 4.9, 4.10 e 4.11 pode ser observado que temas idênticos receberam valores de similaridade relativamente elevados. O limite inferior de similaridade para temas idênticos foi 0.50, decorrente do peso atribuido à similaridade de atributos, sendo este valor gradativamente aumentado em decorrência da similaridade das suas vizinhanças semânticas. $Temas\_isolados$ idênticos atingiram um valor de similaridade igual a 1.0.

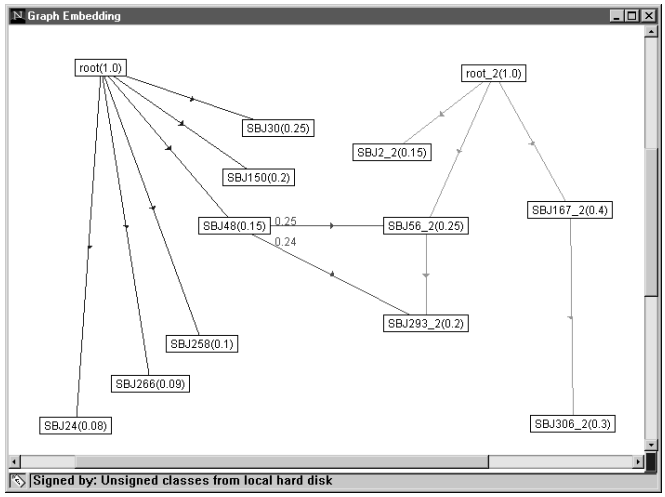

Figura 4.7: Comparação do perfil do usuário $UserONE$ com o perfil do usuário $UserTWO$ levando em conta os $temas\_isolados$.

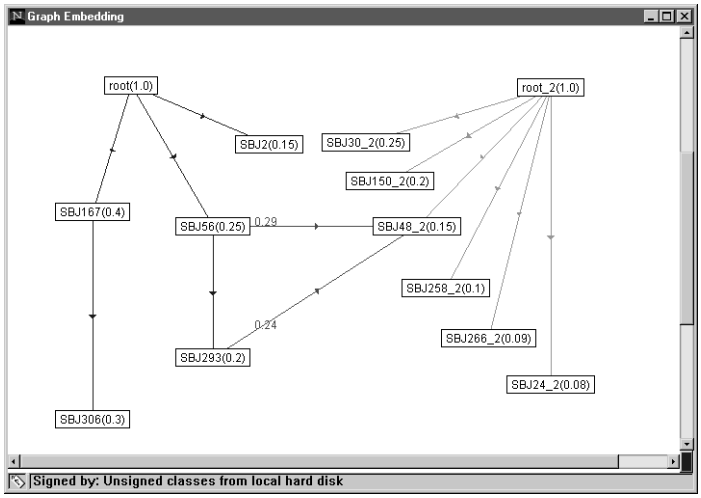

Figura 4.8: Comparação do perfil do usuário $UserTWO$ com o perfil do usuário $UserONE$ levando em conta os $temas\_isolados$.

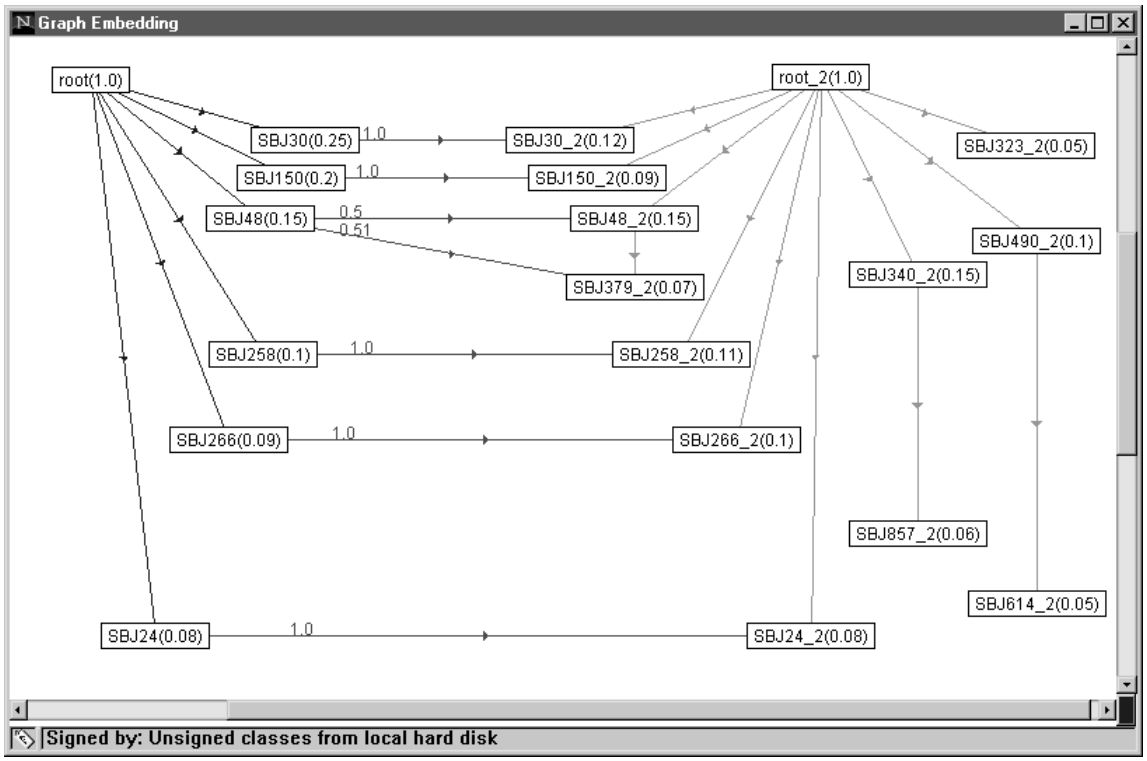

Figura 4.9: Saída gráfica da comparação do perfil do usuário $UserONE$ com o perfil do usuário $UserTHREE$ levando em conta os *temas_isolados*.

Tema de UserONE	**Tema de UserTHREE**	**Similaridade Calculada**	*Relação direta*
SBJ30	*SBJ30_2*	1.0	Atributos idênticos
SBJ150	*SBJ150_2*	1.0	Atributos idênticos
SBJ48	*SBJ48_2*	0.5	Atributos idênticos
SBJ258	*SBJ258_2*	1.0	Atributos idênticos
SBJ266	*SBJ266_2*	1.0	Atributos idênticos
SBJ24	*SBJ24_2*	1.0	Atributos idênticos
SBJ48	*SBJ379_2*	0.51	Dos 8 atributos de *SBJ48* e 4 de *SBJ379_2*, 3 são comuns

Tabela 4.4: Relações entre temas dos perfís dos usuários UserONE e UserTHREE.

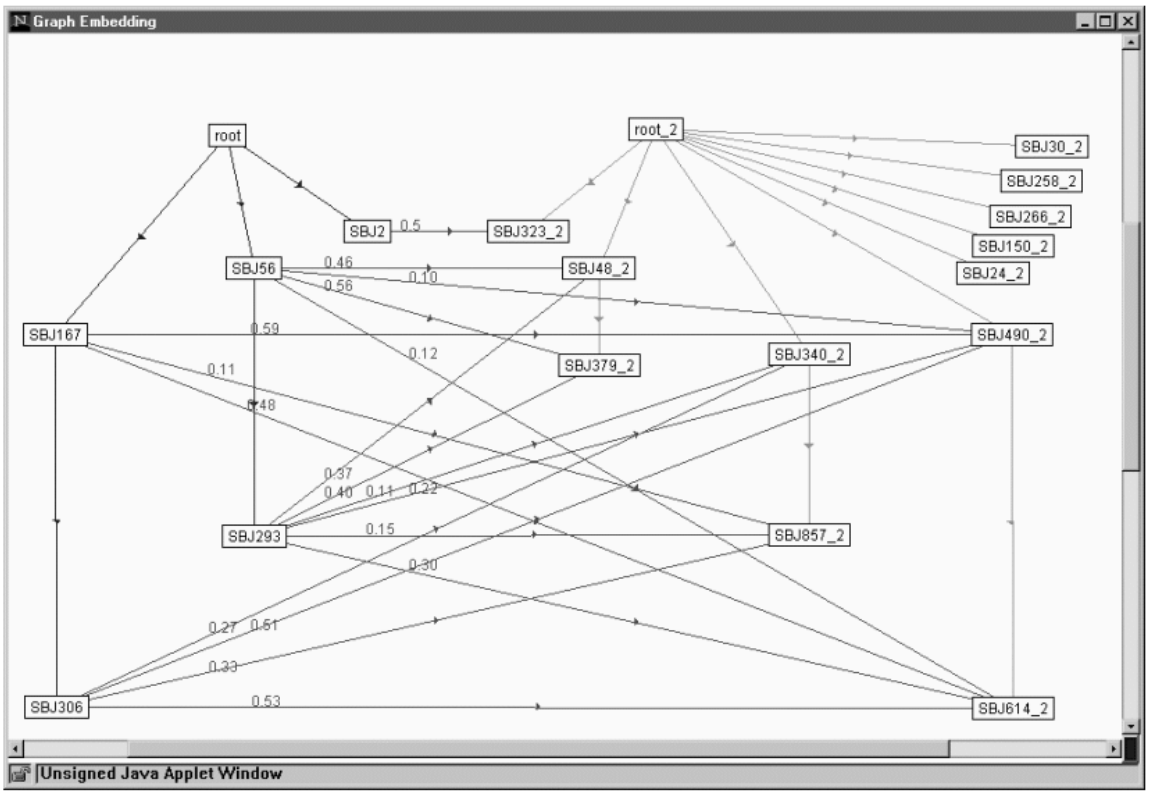

Figura 4.10: Saída gráfica da comparação do perfil do usuário $UserTWO$ com o perfil do usuário $UserTHREE$ sem levar em conta os *temas_isolados*.

Tema de UserTWO	**Tema de UserTHREE**	**Similaridade Calculada**	*Relação direta*
SBJ2	*SBJ323_2*	0.5	Atributos idênticos
SBJ56	*SBJ379_2*	0.56	Atributos idênticos
SBJ167	*SBJ490_2*	0.59	Atributos idênticos
SBJ293	*SBJ614_2*	0.30	Os 4 atributos de *SBJ614_2* formam um subconjunto dos 9 atributos de *SBJ293*

Tabela 4.5: Algumas das relações existentes entre temas dos perfís dos usuários UserTWO e UserTHREE.

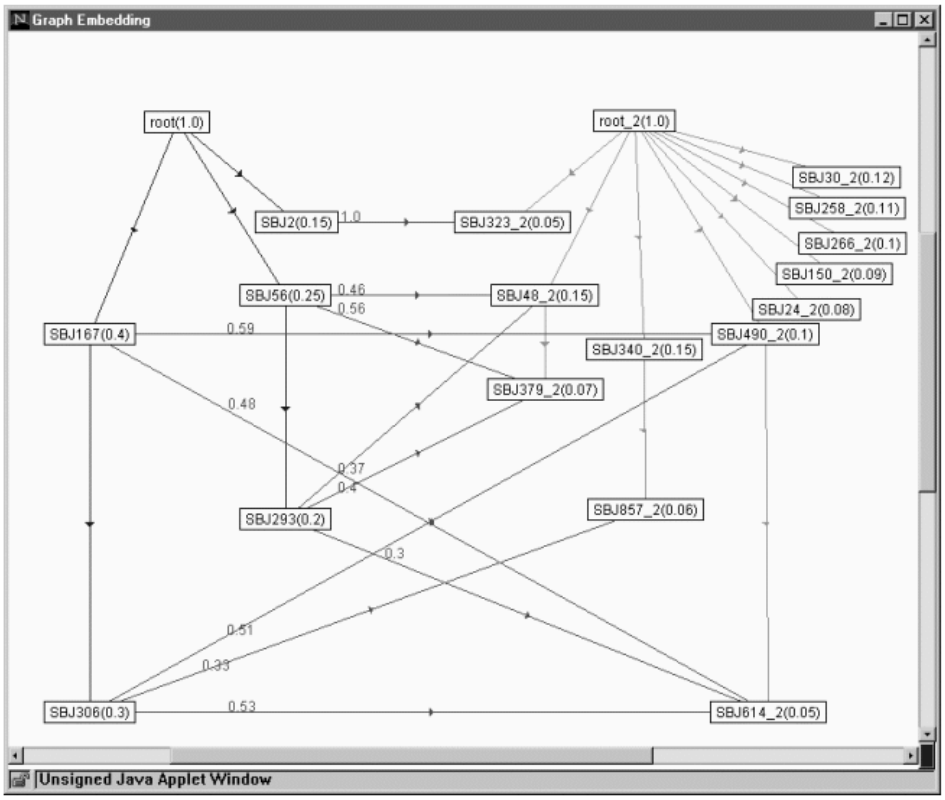

Figura 4.11: Comparação do perfil do usuário $UserTWO$ com o perfil do usuário $UserTHREE$ levando em conta os $temas\_isolados$. Para maior clareza aparecem apenas as ligações decorrentes de valores de similaridade com valor mínimo de 0.30.

4.3.3 Comparação levando em conta o grau de preferência de um tema

O seguinte passo foi efetuar a comparação de um tema $t_i^{perfil_1}$, pertencente ao $perfil_1$ de um usuário $usuário_1$, com um tema $t_j^{perfil_2}$ pertencente ao perfil $perfil_2$ de outro usuário $usuário_2$ levando em conta o grau de preferência $pref_{t_i^{perfil_1}}$ (vide a seção 4.2) do usuário $usuário_1$ em relação ao tema $t_i^{perfil_1}$.

O cálculo do valor da similaridade é executado segundo a fórmula:

$$S_{pref}(t_i^{perfil_1}, t_j^{perfil_2}) = pref_{t_i^{perfil_1}} * S(t_i^{perfil_1}, t_j^{perfil_2}) \tag{4.1}$$

Onde:

$S_{pref}(t_i^{perfil_1}, t_j^{perfil_2})$ é o valor da similaridade levando em conta a preferência;

$S(t_i^{perfil_1}, t_j^{perfil_2})$ é o valor da similaridade entre os temas $t_i^{perfil^1}$ e $t_j^{perfil_2}$, obtido segundo o processo descrito nas seções anteriores;

$pref_{t_i^{perfil_1}}$, é o grau de preferência do usuário $usuário_1$ em relação ao tema $t_i^{perfil_1}$.

A figura 4.12 mostra os resultados da aplicação da fórmula 4.1 aos valores de similaridade obtidos comparando o tema $SBJ293$ pertencente ao perfil do usuário $UserTWO$ com os temas do perfil do usuário $UserTHREE$. Como pode ser observado, o grau de preferência atua

como uma restrição adicional que pode ser explorada para tornar mais eficiente a comparação de perfís.

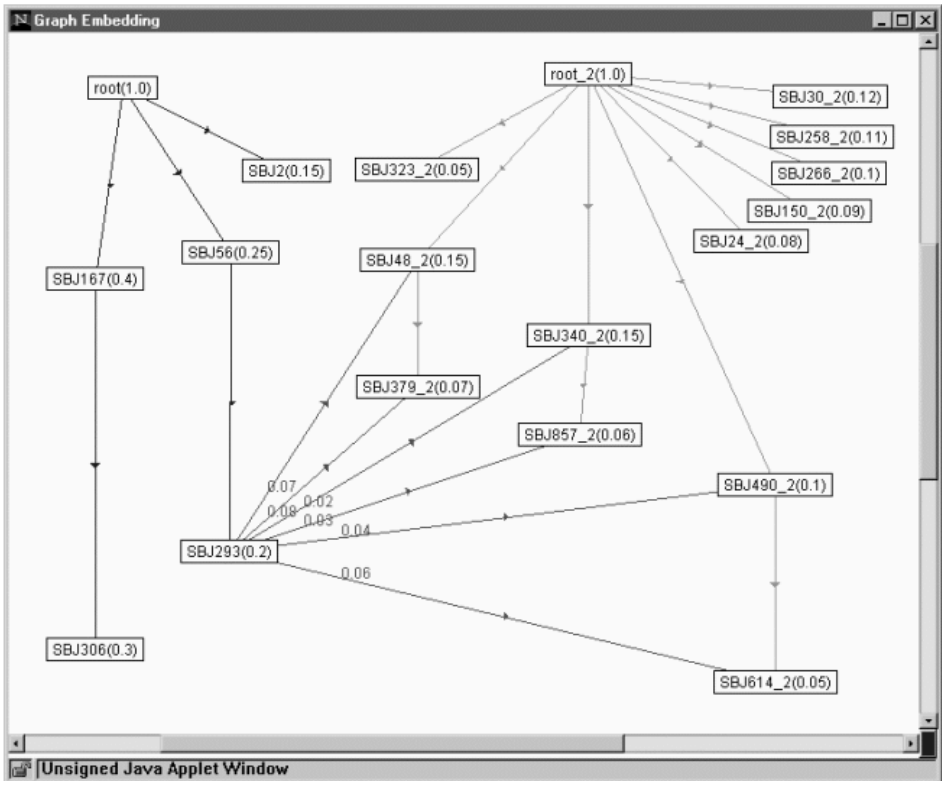

Figura 4.12: Comparação do tema $SBJ293$ do perfil do usuário $UserTWO$ com os temas do perfil do usuário $UserTHREE$ levando em conta o grau de prefência do usuário $UserTWO$ por $SBJ293$, i.e. 0.20.

4.4 Discussão dos resultados e extensões

A principal contribuição deste trabalho foi a de tornar possível uma comparação quantitativa de perfís de usuários a partir da utilização do Agente PersonalSearcher.

No decurso dos experimentos, foram necessários alguns ajustes na aplicação do modelo MD3 como métrica de similaridade entre conceitos pertencentes a perfís diferentes:

- Exclusão do cálculo de similaridade de nomes entre conceitos pois na atual implementação do PersonalSearcher o nome de um tema não carrega informação relevante, vide cap. 3;
- Inclusão dos pesos das palavras usadas como atributos dos temas no cálculo da cardinalidade da interseção e diferença entre os conjuntos de atributos dos conceitos sendo comparados, vide o cap. 3;
- Tratamento diferenciado de *temas_isolados*, vide a seção 4.3.1;
- Inclusão opcional do grau de preferência de um tema no cálculo da similaridade, vide 4.3.3.

Estas alterações mostraram-se apropriadas, na medida em que os resultados obtidos fornecem medidas quantitativas manipuláveis por um agente de interface para decidir as condições

nas quais uma troca de conhecimento pode mostrar-se útil. Caberá ao agente que eventualmente recebe o conhecimento partilhado decidir o destino do mesmo, seja incorporando-o diretamente a sua própria hierarquia ou, de maneira mais cautelosa, conservando-o como uma representação separada de sua própria, existindo apenas ligações que as interconectem.

A comparação de dois perfís como um todo, i.e. o cálculo de um valor de similaridade global entre dois perfís, não foi abordada, isto por considerar-se que um agente de busca primeiramente está interessado na extensão de conceitos particulares (e.g., quando da ampliação de uma consulta com palavras de significados similares) através de sinônimos ou conceitos análogos. Nada impede, porém, que um valor global seja calculado a partir dos resultados particulares acessíveis através da aplicação do presente trabalho.

A partir das experiências realizadas, aparecem outras possíveis direções para trabalhos futuros. Devem ser realizados estudos para verificar qual o grau de adequação dos pesos para a similaridade de atributos e de vizinhança semântica, no momento estabelecidos igualmente como sendo *0.5*.

A identificação de conceitos similares pertencentes a perfís diferentes, como foi visto no capítulo 4, leva a diversas estratégias possíveis de partilha:

- **Partilha de palavras-chave:** quando um agente de interface ag_{id} recebe uma solicitação de auxílio de um outro agente de interface ag_{alter}, contendo os atributos associados a um tema $t_{externo}$, ele pode procurar no seu próprio perfil $perfil_{id}$ o(s) tema(s) que guardam um grau de similaridade superior a um limiar que irá depender do contexto (e.g., considerando como contexto apenas o grau de conhecimento de um certo domínio d_{cog}: qual a crença expressada pelo agente ag_{alter} sobre o seu grau de conhecimento do domínio d_{cog}?, qual a crença do agente ag_{id} em relação ao domínio d_{cog}?)

- **Partilha de subconjuntos de um perfil:** para além das palavras, a partilha neste nível pode revelar informações (contidas na estrutura de um perfil) sobre quais são as relações existentes entre diversos temas. Porém, este tipo de partilha pode implicar uma intromissão maior na privacidade do agente ag_{id}. Para tanto os agentes devem ser capazes de avaliar, além das questões levantadas pela estratégia anterior, outras como: qual a autoridade do agente ag_{alter} sobre ele?, qual a crença do agente ag_{alter} na honestidade do conhecimento eventualmente fornecido pelo agente ag_{id}? ;

- **Partilha de documentos agrupados sob um tema:** esta estratégia, explorada e.g. em [Williams and Ren, 2001] sem utilizar um contexto organizacional, agudiza ainda mais a questão da privacidade, pois os próprios documentos do usuário são expostos. Isto não seria nada preocupante (sendo até desejável) no caso de uma especificação de projeto de software que deva ser partilhada num grupo de trabalho. Por outro lado, tal partilha seria inadmissível no caso de um arquivo pessoal de *e-mail* relatando problemas econômicos.

- **Combinação de estratégias:** fornecendo, por exemplo, um conjunto de documentos e o tema que os identifica.

É fácil perceber, pelos casos acima, que o *contexto organizacional* da partilha deve ser considerado, e que este influencia diretamente a escolha da estratégia de partilha mais apropriada. Para definir o contexto organizacional, podem ser usados modelos organizacionais de SMA [Hannoun et al., 2000, Hübner et al., 2002] em conjunção com técnicas que permitam acessar subconjuntos de um perfil levando em conta, por exemplo, os papéis específicos (e.g. empregado numa firma, pai, integrante de um grupo de pessoas com um hobby em comum, etc.) que um usuário desempenha quando executa uma busca.

Adicionalmente, o algoritmo de comparação, que realiza um percurso completo em largura, deve ser explorado em perfís de usuários maiores que os disponíveis atualmente para

avaliar o seu desempenho [3]. Neste ponto, o contexto organizacional pode ser usado como chave para podar o espaço de busca de um conceito dentro da ontologia de um usuário como opção para melhorar o desempenho. Neste caso, um agente de interface como o PersonalSearcher deverá ter a sua capacidade de raciocínio extendida de forma a poder tirar proveito dessas informações.

Quanto à implementação das estratégias de partilha, elas podem ser implementadas como protocolos de interação (como em [Lugo et al., 2001]) que levem em conta o contexto organizacional.

Ainda outra opção para futuras experiências envolve a comparação de hierarquias de agentes individuais com a de grupos de agentes e/ou com ontologias mais gerais tais como Wordnet [Miller et al., 1993].

[3] O tempo de execução das comparações foi de poucos segundos, mas não pode ser esquecido o fato dos perfís disponíveis serem de tamanhos reduzidos.

Capítulo 5

Conclusões

Os resultados atingidos demonstraram não somente ser factível comparar o conhecimento de diversos perfís de usuário, na forma em que eles são representados e tratados pelo PersonalSearcher, mas também sugerem a promessa de melhorar efetivamente o desempenho de agentes individuais de busca, através da partilha de conhecimento decorrente de experiências de outros agentes.

Um ponto sensível diz respeito à dependência direta das hierarquias disponíveis de algoritmos de cunho similar ao implementado nesta abordagem. Como neste caso o PersonalSearcher obtém de forma automática estas hierarquias, as mesmas não são comparáveis àquelas obtidas com a intervenção humana, estabelecendo um teto natural à eficácia da partilha.

Outro aspecto que não pode ser negligenciado é a necessidade de atrelar a uma aplicação de SMA um modelo organizacional, não apenas com a visão que a preconiza como fundamento de coordenação, mas para tirar proveito efetivo das informações que podem ser adquiridas. Assim, organização e partilha de conhecimento, assim como nas sociedades humanas, são dois aspectos interdependentes em sistemas multiagentes voltados para a recuperação de informações.

Referências Bibliográficas

[Chaffee and Gauch, 2000] Chaffee, J. and Gauch, S. (2000). Personal ontologies for web navigation. In *CIKM*, pages 227–234.

[Finin et al., 1998] Finin, T., Nicholas, C., and Mayfield, J. (1998). Agent-based information retrieval. In *IEEE ADL'98, Advances in Digital Libraries Conference '98.*

[Fridman and Musen, 1999] Fridman, N. and Musen, M. A. (1999). An algorithm for merging and aligning ontologies: Automation and tool support. In *Proc. of the Workshop on Ontology Management at the AAAI-99 Conf.* AAAI Press.

[Godoy, 2001] Godoy, D. (2001). *PersonalSearcher: Un Agente Inteligente para Búsqueda de Información en la WWW. Tesis de Magister.* Universidad Nacional del Centro de la Provincia de Buenos Aires, Tandil, Argentina.

[Godoy and Amandi, 2000] Godoy, D. and Amandi, A. (2000). PersonalSearcher: An Intelligent Agent for Searching Web Pages. In *Proc. of the Intl. Joint Conference, 7th Ibero-American Conference on AI, 15th Braziliam Symposium on AI. LNAI 1952.* M. C. Monard and J. S. Sichman editors.

[Gruber, 1995] Gruber, T. (1995). Towards principles for the design of ontologies used for knowledge sharing. *Intnl. Journal of Human and Computer Studies*, 43(5/6):907–928.

[Guarino, 1997] Guarino, N. (1997). Formal ontological distinctions for information organization, extraction, and integration. In *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology. LNAI 1299.* M. Pazienza(ed). Springer.

[Guarino and Giaretta, 1995] Guarino, N. and Giaretta, P. (1995). Ontologies and knowledge bases- towards a terminalogical clarification. In *N. Guarino and P Giaretta. Ontologies and knowledge bases- towards a terminological clarification.* N.J. Mars, ed., Towards Very Large Knowledge Bases - Knowledge Building and Knowledge Sharing 1995. IOS PRESS.

[Hannoun et al., 2000] Hannoun, M., Boissier, O., and Sichman, J. S. (2000). MOISE: an organizational model for multi-agent systems. In *Proc. of the Intl. Joint Conference, 7th Ibero-American Conference on AI, 15th Braziliam Symposium on AI. LNAI 1952.* M. C. Monard and J. S. Sichman editors.

[Hübner et al., 2002] Hübner, J. F., Sichman, J. S., and Boissier, O. (2002). A Model for the Structural, Functional and Normative Specification of a MAS Organization. In *AAMAS-2002*.

[Jennings and Wooldridge, 1998] Jennings, N. and Wooldridge, M. (1998). Applications of intelligent agents. In *Agent Technology: Foundations, Applications and Markets.* Jennings, N. and Wooldrigde, M.(eds.). Springer Verlag.

[Klusch, 2001] Klusch, M. (2001). Intelligent information agents. In *Third European Agent Systems Summer School. Advanced Course on Artificial Intelligence ACAI-01.*

[Kobayashi and Takeda, 2000] Kobayashi, M. and Takeda, K. (2000). Information retrieval on the web. *ACM Computing Surveys*, 32(2):144–173.

[Krumhansl, 1978] Krumhansl, C. (1978). Concerning the Applicability of Geometric Models to Similarity Data: The Interrelationship Between Similarity and Spatial Density. *Psychological Review*, 85(5):445–463.

[Levy and Weld, 2000] Levy, A. Y. and Weld, D. S. (2000). Intelligent internet systems. *Artificial Intelligence*, 118(1-2):1–14.

[Lugo et al., 2001] Lugo, G. G., Hübner, J. F., and Sichman, J. S. (2001). Representação e Evolução de Esquemas Sociais em SMA: um enfoque funcional. In *Anais do Encontro Nacional de Inteligência Artificial 2001.* Fortaleza, Ceará, Brasil. SBC.

[Maes, 1994] Maes, P. (1994). Agents that Reduce Work and Information Overload. *Communications of the ACM*, 37(7):30–40.

[McGuinness et al., 2000] McGuinness, D. L., Fikes, R., Rice, J., and Wilder, S. (2000). An Environment for Merging and Testing Large Ontologies. In *Proc. of the 7th Intnl. Conf. on Principles of Knowledge Representation and Reasoning (KR2000).* Colorado, USA.

[Miller et al., 1993] Miller, G., Beckwith, R., Felbaum, C., Gross, D., and Miller, K. (1993). *Introduction to WordNet: An Online Lexical Database.* http://www.cogsci.princeton.edu/ wn/.

[Mitra et al., 2000] Mitra, P., Wiederhold, G., and Kersten, M. (2000). A graph oriented model for articulation of ontology interdependendies. In *VII Conference on Extending Database Technology - EDBT 2000.*

[Mladenic, 1999] Mladenic, D. (1999). Text-learning and related intelligent agents: a survey.

[Pretschner, 1999] Pretschner, A. (1999). *Ontology Based Personalized Search. MSc. Thesis.* University of Kansas, USA.

[Rodríguez, 2000] Rodríguez, M. A. (May, 2000). *Assessing semantic similarity among spatial entity classes. PhD. Thesis.* University of Maine, USA.

[Tversky, 1977] Tversky, A. (1977). Features of Similarity. *Psychological Review*, 84(4).

[Wache et al., 2001] Wache, H., Vögele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H., and Hübner, S. (2001). Ontology-based integration of information - a survey of existing approaches. In *Proc. of the IJCAI 2001 Workshop on Ontologies and Information Sharing*.

[Williams and Ren, 2001] Williams, A. B. and Ren, Z. (2001). Agents teaching agents to share meaning. In *Proc. of the 5th Intl. Conf. on Autonomous Agents*. ACM.