Diogo Santana Martins

Uma abordagem para recuperação de informações sensível ao contexto usando retroalimentação implícita de relevância

São Carlos – SP

Julho de 2009

Diogo Santana Martins

Uma abordagem para recuperação de informações sensível ao contexto usando retroalimentação implícita de relevância

Orientador:
Prof. Dr. Mauro Biajiz

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
DEPARTAMENTO DE COMPUTAÇÃO
UNIVERSIDADE FEDERAL DE SÃO CARLOS

São Carlos – SP

Julho de 2009

Ficha catalográfica elaborada pelo DePT da Biblioteca Comunitária da UFSCar

M386ua

Martins, Diogo Santana.
Uma abordagem para recuperação de informações sensível ao contexto usando retroalimentação implícita de relevância / Diogo Santana Martins. -- São Carlos : UFSCar, 2010.
108 f.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2009.

1. Recuperação da informação. 2. Ciência de contexto. 3. Expansão de consultas. 4. Personalização. I. Título.

CDD: 005.74 (20ª)

Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia

Programa de Pós-Graduação em Ciência da Computação

"Recuperação de informações sensível ao contexto usando retroalimentação implícita de relevância"

DIOGO SANTANA MARTINS

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. Marilde Terezinha Prado Santos
(DC/UFSCar)

Prof. Dr. Antonio Francisco do Prado
(DC/UFSCar)

Prof. Dr. Marina Teresa Pires Vieira
(UNIMEP)

São Carlos
Agosto/2009

Agradecimentos

Agradeço à minha família pelo apoio — em especial à minha mãe, Arlete Santana, por acreditar no meu ingresso na carreira acadêmica; e pela paciência em ouvir minhas lamúrias nos momentos difíceis, aconselhando-me sempre. Agradeço a Mario Liziér por ter tornado essa jornada mais fácil de ser percorrida. Registro também agradecimentos aos colegas de laboratório pelas proveitosas discussões, resenhas e colaborações, em específico Luiz Santana, Gustavo Afonso, Rafael Miani e Raphael Melo. Agradeço aos professores Antonio Prado e Wanderley Souza, pelas orientações e pelas oportunidades de pesquisa; e à Profa. Marilde Santos, pelo respaldo nos momentos finais desse projeto.

Agradeço ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo financiamento da minha pesquisa; ao PPG-CC DC-UFSCar pelo respaldo administrativo e pelo auxílio financeiro para participação em congressos; à coordenação do curso de medicina da UFSCar pela oportunidade de conduzir estudos de caso junto ao grupo piloto de avaliação do Portfólio Reflexivo Eletrônico; e à coordenação da Universidade Aberta do Brasil da UFSCar (UAB-UFScar), especialmente aos Profs. Joice Otsuka, Sandra Abib e Roberto Ferrari pela oportunidade de avaliar a pesquisa junto aos dados de educação a distância da UAB-UFSCar.

Agradecimentos especiais dedico ao meu orientador Mauro Biajiz (*in memoriam*), o qual não assistiu em vida ao desfecho dessa pesquisa. Agradeço por ter-me apresentado ao mundo da pesquisa, durante iniciação científica e mestrado. Por acreditar e investir em minhas ideias, mesmo quando eu tentava descartá-las. Por ter proporcionado o privilégio de tê-lo como orientador e a oportunidade de trabalhar com um pesquisador ético, competente e equilibrado; por ter sido um modelo de profissional que sempre lembrarei e terei como referência ao longo da minha carreira.

Resumo

Esta dissertação reporta a abordagem RISC-RIR (Recuperação de Informações Sensível ao Contexto usando Retroalimentação Implícita de Relevância) para melhorar a qualidade dos resultados de sistemas de recuperação de informação que podem ser acessados via dispositivos móveis. Tais dispositivos podem impor aos usuários dificuldades em expressar consultas precisas e contextualizadas, assim como dificuldades para percorrer listas longas de resultados. A abordagem RISC-RIR provê recuperação personalizada de informação por meio da integração do contexto de trabalho dos usuários em um mecanismo para retroalimentação implícita de relevância. Para alcançar tal objetivo, a abordagem inclui uma arquitetura para gerenciar a transformação e processamento de informações de contexto que são usadas para guiar a expansão de consulta. O projeto foi avaliado em dois estudos de caso e os resultados revelam que a abordagem RISC-RIR melhora a qualidade da recuperação de informação. Os ganhos observados permanecem expressivos mesmo com variações na magnitude da coleção de documentos, da diversidade de usuários e de situações de contexto.

Abstract

This dissertation reports an approach to enhance information retrieval systems which are accessed through mobile devices. This sort of devices can impose constraints on user-computer interaction, mainly concerning expression of contextualized queries and navigation of long lists of results. Our approach integrates user work context in implicit relevance feedback, which is developed over a Case Based Reasoning methodology, with the purpose of providing personalized information retrieval. To tackle these issues, it was developed an architecture to manage transformation and processing of context information, as well as selection of evidences to expand queries through implicit relevance feedback. The project was evaluated on two case studies and the obtained results show that our approach enhances the quality of information retrieval, even under varitions on the size of the document collection, on the diversity of users and on context situations.

Sumário

Lista de Figuras

1 Introdução

Esta dissertação desenvolve uma abordagem para melhorar a qualidade dos resultados de sistemas de recuperação de informação que podem ser acessados via dispositivos móveis. A utilização de dispositivos móveis com recursos restritos (telas pequenas e teclados limitados) como meio de acesso impõe dificuldades ao uso de sistemas de recuperação de informação, devido principalmente a limitações para entrada e visualização das informações. Estas limitações configuram-se na dificuldade em expressar consultas precisas e contextualizadas bem como em dificuldade para percorrer listas longas de resultados.

Devido a estas limitações, é importante que o sistema de recuperação de informação forneça documentos relevantes no topo da lista de resultados, mesmo frente a consultas curtas e pouco precisas, isto é, que o sistema forneça um bom atendimento às necessidades informacionais do usuário. Estudos em comportamento de busca por informação revelam que uma necessidade informacional é fortemente influenciada pelo contexto no qual o usuário está inserido [Ingwersen and Järvelin 2005]. Embora o contexto de uma necessidade informacional possa ser interpretado sob diferentes níveis de abstração, uma interpretação que tem sido usada na literatura é considerar contexto como atributos da tarefa em que o usuário está envolvido enquanto busca informação (e.g. ferramentas usadas, documentos lidos, comunicações com outros usuários) [Byström and Hansen 2005].

Com vistas à exploração desses atributos de contexto durante a recuperação de informação, a abordagem desenvolvida nesta dissertação fornece resultados de busca personalizados ao contexto da tarefa de trabalho do usuário. O restante deste capítulo está organizado do seguinte modo: a seção 1.1 expõe os principais fatores que motivaram esta pesquisa; a seção 1.2 expõe brevemente os objetivos do trabalho desenvolvido e os resultados obtidos; por fim, a seção 1.3 delineia a organização da dissertação.

1.1 Motivação

O trabalho relatado nesta dissertação foi inspirado em problemas de recuperação de informação observados em ambientes de computação ubíqua para apoio a sistemas de trabalho colaborativo, em particular ambientes de aprendizado ubíquo. Aprendizado Colaborativo Assistido por Computador (*Computer Supported Collaborative Learning* ou CSCL) é um ramo do Trabalho Colaborativo Assistido por Computador (*Computer Supported Collaborative Work* ou CSCW) que trata do emprego de tecnologias de informação e comunicação na execução de processos educacionais em grupo. Sistemas de CSCL, também conhecidos como Ambientes de Aprendizado Eletrônico, oferecem uma plataforma para que os estudantes possam aprender em grupo, através de comunicação e troca de informações, além de facilitar a reflexão do aluno durante seu aprendizado. Estes objetivos são atingidos via a coordenação de ferramentas para diferentes propósitos, como videoconferência, comunicadores instantâneos, correio eletrônico, listas de distribuição de e-mails e fóruns de discussão. Em vista da grande disponibilidade de recursos para se criar e publicar conteúdos em plataformas de aprendizado eletrônico, um requisito importante nesses ambientes são meios para organizar e recuperar esses conteúdos.

Uma especialização do Aprendizado Eletrônico é o Aprendizado Ubíquo (*Ubiquitous Learning* ou simplesmente *UbiLearning*), que objetiva conciliar os avanços de Computação Ubíqua e CSCL. Em ambientes de *UbiLearning*, tecnologias de informação e comunicação e pequenos dispositivos móveis são largamente empregados, equipando os estudantes com mobilidade e ferramental para empreender tarefas de aprendizado cooperativamente e de forma distribuída, a qualquer momento. Consequentemente, recursos são providos para aprender a coisa certa, no tempo certo, da forma correta [Ogata and Yano 2004].

A introdução de dispositivos móveis (como *tablets*, PDAs e *smart phones*) em ambientes educacionais impõe diversas restrições na interação usuário-computador, principalmente devido aos recursos limitados desses dispositivos, tais como pequenas telas, funcionalidades restritas para entrada de dados, pouca largura de banda e conexão intermitente de rede, entre outras. Tais restrições tornam-se mais críticas quando o usuário está interagindo com um sistema de Recuperação de Informação (RI) baseado em palavras-chave, que fortemente depende da entrada de termos de busca adequados e em número suficiente e frequentemente retorna uma grande quantidade de resultados a navegar.

Pesquisas acerca do comportamento de usuários de Internet [Jansen, Bernard J. and Pooch, Udo 2001] revelam que usuários de sistemas de RI em geral provêm poucas chaves de busca e dispendem a maioria do tempo de interação navegando longas listagens dos resultados em busca dos documentos realmente relevantes. Este é um sintoma de que a tradução da necessidade informacional em uma consulta pode conduzir a uma representação equivocada das intenções originais do usuário, devido à cobertura parcial dos dois parâmetros que caracterizam uma necessidade informacional: o tema, expresso pela consulta do usuário; e o contexto, que determina por quê a informação está sendo buscada e como a informação será posteriormente empregada [Hernandez et al. 2007].

Apesar do conhecimento do contexto de trabalho apresentar potencial para melhorar a qualidade dos resultados da busca, pode tornar-se inviável enumerar e detectar todas as possíveis configurações de contexto de trabalho em que um usuário participa num fluxo de trabalho genérico [Freund and Toms 2005]. Uma possível alternativa para atacar essa limitação é de início delimitar o domínio em que o contexto de trabalho será capturado e aproximar a situação atual do usuário usando atributos indiretos da tarefa de trabalho que usuário o está executando. Mais especificamente, captura-se atributos que descrevem o comportamento do usuário enquanto realiza buscas (e.g. que documentos estão sendo navegados, quais são as tarefas agendadas para acontecer nesse momento, com quem o usuário está interagindo enquanto executa a tarefa, que ferramentas está empregando, e assim por diante). Tais atributos indiretos podem ser coletados de várias fontes presentes no ambiente eletrônico de trabalho, por exemplo a partir da monitoria de aplicações e documentos que o usuário emprega ao executar suas atividades.

Uma vez que o sistema de RI torne-se capaz de representar e processar atributos do contexto de trabalho, esses meta-dados podem ser usados como evidências para contextualizar as consultas. Uma técnica adequada para atacar esse problema é o modelo de expansão de consulta denominado retroalimentação de relevância (*relevance feedback*) [Ruthven and Lalmas 2003]. As abordagens para retroalimentação de relevância diferenciam-se principalmente no tipo de evidências empregadas. Na modalidade explícita de retroalimentação de relevância (*explicit relevance feedback*) o usuário submete uma consulta, recebendo um conjunto de resultados em resposta, dos quais conscientemente aponta quais julga serem relevantes; na modalidade implícita de retroalimentação de relevância (*implicit relevance feedback*), o usuário não interfere no processo de expansão, de forma que as evidências são coletadas a partir de fontes indiretas, como

logs de consulta, monitoramento de cliques, etc; uma terceira modalidade, a modalidade cega de retroalimentação de relevância (*blind relevance feedback* ou *pseudo relevance feedback*) toma como evidência um subconjunto de documentos que aparecem no topo da lista de resultados. Em todas essas abordagens, os termos dos documentos que são indicados como relevantes são extraídos e filtrados para posteriormente expandir a consulta original.

Em particular, retroalimentação implícita de relevância tem sido aplicada para explorar o contexto de necessidades informacionais, com base em evidências da interação direta do usuário com os documentos, por meio do rastreio de documentos navegados em uma sessão de busca [Shen et al. 2005a] [Teevan et al. 2005] [Jung et al. 2007a]. Concomitantemente, tem sido comum a defesa de estratégias mais abrangentes para representar o contexto do usuário [Hernandez et al. 2007], que considerem não apenas a interação direta com os documentos, mas também a interação com o ambiente eletrônico de trabalho em que os documentos estão encerrados [Freund and Toms 2005] [Redon et al. 2007].

Com base nas diretrizes apontadas, esta dissertação desenvolve uma abordagem de retroalimentação de relevância baseada no contexto de trabalho do usuário. A próxima seção provê uma visão geral dos principais objetivos e resultados deste projeto.

1.2 Objetivos e resultados

Esta dissertação integra o contexto de trabalho dos usuários em um mecanismo para retroalimentação implícita de relevância, de forma a prover recuperação personalizada de informação. Para alcançar tal objetivo, a abordagem define uma arquitetura para gerenciar a transformação e processamento de informações de contexto, e seleção de evidências para expansão de consultas via retroalimentação implícita de relevância.

A operacionalização da personalização foi obtida por meio da técnica de retroalimentação implícita de relevância (*implicit relevance feedback*), uma técnica automática e transparente de expansão de consulta. Nesta técnica, a expansão ocorre baseada na extração de termos expressivos de documentos (evidências) que implicitamente foram julgados como relevantes. Na abordagem desenvolvida, para selecionar as evidências foi utilizado Raciocínio Baseado em Casos sobre o contexto de trabalho do usuário. Esta forma de raciocínio permite trabalhar com a premissa de que se um documento mostrou-se útil em uma configuração particular de atributos de contexto, ou seja, em

uma situação particular, o mesmo documento pode ser uma potencial fonte de relevância em situações similares.

Nesse ínterim, o contexto de trabalho é modelado usando uma ontologia de contexto que serve como base para registrar as situações em que o usuário interage com o sistema. Durante o processo de recuperação dos documentos, as situações associadas aos documentos são comparadas à situação atual do usuário, e esta comparação fornece documentos contextualmente relacionados que são usados como fontes de evidência para expandir consultas.

Foram desenvolvidos dois estudos de caso para avaliar a abordagem, em ambientes distintos de aprendizado eletrônico e com diferentes quantidades de documentos, registros de interação e diversidade de usuários. Considerando as coleções de teste obtidas, para ambos os estudos de caso os resultados revelam que a abordagem desenvolvida apresenta ganhos em precisão nos níveis mais baixos de revocação. Ou seja, quando comparada aos sistemas de referência, a abordagem desenvolvida retorna mais documentos relevantes no topo da lista de resultados. Tais comportamentos trazem benefícios para os usuários de forma geral, e para os usuários de dispositivos móveis, de modo particular, pois os ganhos em precisão tendem a diminuir a necessidade de reformulação de consultas, navegação e rolagem de tela, uma vez que mais documentos relevantes são apresentados no topo da lista de resultados.

Expostos os principais objetivos e resultados trabalhados nesta dissertação, a próxima seção delineia a organização geral dos capítulos subsequentes.

1.3 Organização da dissertação

O restante desta dissertação está organizado da seguinte forma:

Capítulo 2 Apresenta o tema de Recuperação de Informação, expondo suas tarefas típicas e as variações de estrutura dos documentos com os quais tais tarefas podem lidar. São abordados os principais modelos, com ênfase no modelo de espaço de vetores. Por fim, apresentam-se as principais abordagens para avaliação de sistemas de Recuperação de Informação.

Capítulo 3 Caracteriza a importância do conceito de contexto em Recuperação de Informação, expondo as principais tendências de pesquisa para lidar com este pro-

blema. É cedida ênfase especial às pesquisas de contexto relacionadas à Computação Ubíqua, Personalização e Recuperação Interativa de Informação.

Capítulo 4 Introduz o tema de expansão de consulta como otimização aos processos de Recuperação de Informação. Nesse sentido, são expostos os diferentes tipos de modificação de consulta, enfatizando os métodos baseados em retroalimentação de relevância.

Capítulo 5 Trata da abordagem RISC-RIR. São fornecidos detalhes dos repositórios da abordagem, estratégias para gerenciamento dos dados de contexto, o processo de seleção de evidências para retroalimentação de relevância e o método de expansão de consulta.

Capítulo 6 Apresenta os estudos de casos que foram desenvolvidos em dois ambientes de aprendizado eletrônico.

Capítulo 7 Discute trabalhos correlatos ao apresentado nesta dissertação.

Capítulo 8 Conclui a dissertação e direciona trabalhos futuros.

1.4 Considerações finais

Este capítulo introduziu o tema desta dissertação e delimitou seu escopo, apresentando as motivações e temas de pesquisa envolvidos. O próximo capítulo introduz o tema de Recuperação de Informação.

2 *Recuperação de Informação*

Desde o advento dos primeiros grandes repositórios de documentos — as bibliotecas — os problemas relacionados a armazenamento e localização de informações são alvo de investigação científica. Grandes avanços científicos e tecnológicos em métodos de acesso a informações consolidaram-se desde então, acompanhando as demandas da sociedade por aquisição e produção de conhecimento. Contudo, foi inicialmente com a crescente disponibilidade de bases de documentos de texto completo nas bibliotecas e, posteriormente, com o vertiginoso crescimento do volume de informações na Internet, que os sistemas de recuperação de informação disseminaram-se em larga escala, tornando-se ferramentas atualmente imprescindíveis ao tratamento da enchente de informações que inunda todas as áreas do conhecimento humano.

Este capítulo introduz o tema de Recuperação de Informação, ressaltando modelos, processos e estratégias de avaliação. Seu conteúdo está organizado da seguinte forma: a seção 2.1 apresenta Recuperação de Informação como linha de pesquisa, expondo as tarefas típicas de seus processos; a seção 2.2 trata dos modelos de recuperação de informação, com ênfase no modelo de espaço de vetores; a seção 2.3 introduz técnicas de avaliação de sistemas de recuperação de informação; por fim, a seção 2.4 apresenta considerações finais para concluir este capítulo.

2.1 Tarefas de um processo de recuperação de informação

Como linha de pesquisa, Recuperação de Informação (ou simplesmente RI) preocupa-se com a representação, organização e acesso de objetos informacionais de natureza não-estruturada. A representação e a organização desses objetos são articulados por processos e fundamentados por modelos, de tal forma a prover, ao usuário, fácil acesso a subconjuntos específicos que satisfaçam sua necessidade informacional [Baeza-Yates and Ribeiro-Neto 1999]. A figura 2.1 apresenta como coordenam-se as tarefas típicas

de um processo de recuperação de informação.

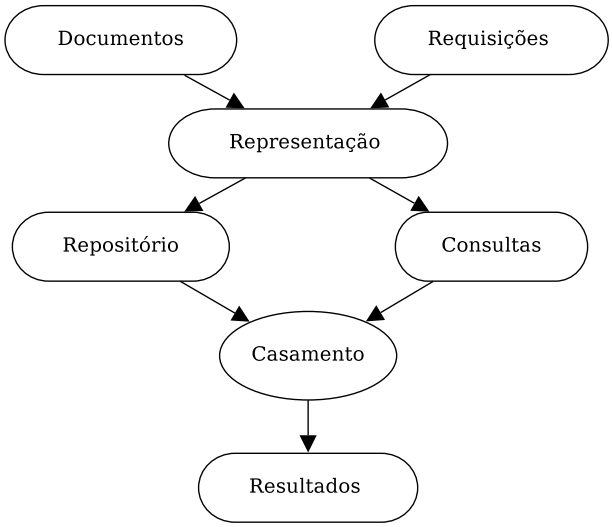

Figura 2.1: Coordenação das tarefas típicas de um processo de RI. Adaptado a partir de [Baeza-Yates and Ribeiro-Neto 1999, Ingwersen and Järvelin 2005]

A preocupação central desse processo são os documentos, objetos cujo conteúdo informacional é não-estruturado, ou seja, compõem-se por estruturas informacionais irregulares e difíceis de ser abstraídas em modelos de dados. Será adotada a convenção de que *documentos* designam objetos informacionais cujo conteúdo é composto majoritariamente por texto-livre, sejam eles não-estruturados ou semi-estruturados. O conjunto de documentos, assim definidos, tratados por um sistema de recuperação de informação (ou simplesmente sistema de RI), é denominado coleção ou *corpus*. Embora a Recuperação de Informação lide também com outras classes de informações não-estruturadas, como conteúdos audiovisuais, essas classes demandam abordagens específicas que não serão abordadas, pois estão fora do escopo desta dissertação.

Devido às estruturas informacionais irregulares dos documentos, sistemas de RI extraem de cada documento uma representação intermediária, capaz de tornar seu vocabulário mais compacto e de preservar a expressividade de seus tópicos. Essa representação intermediária é mantida juntamente com o documento em um repositório.

Na literatura de Recuperação de Informação, a estratégia mais difundida de representação dos documentos dá-se pela indexação de seus termos. Os termos são colecionados ao longo de um processo de indexação que pode ser tanto intelectual (ou manual, pois é aplicado por um agente humano) quanto automático (quando aplicado por um algoritmo). Quando o processo de indexação é intelectual, as chaves de indexação provêm de alguma fonte de vocabulário controlado, como taxonomias, *thesauri* ou ontologias; quando é automático, embora fontes de vocabulário controlado também

possam ser empregadas, é mais comum que as chaves provenham do conteúdo em si e sejam determinadas por algoritmos para processamento de texto.

O outro viés do processo são as requisições de informação, que expressam necessidades informacionais dos usuários. Uma necessidade informacional emerge de uma lacuna, conscientemente identificada, no conhecimento do usuário, responsável por disparar a busca por informação [Borlund 2003b]. A busca por informação, por sua vez, inspira a formulação de uma requisição de informação, que pode assumir diversas formas, a exemplo de uma expressão semi-estruturada, uma expressão em linguagem natural, um exemplo do conteúdo desejado, entre outras variantes.

Independentemente da forma que a requisição de informação assumir, ela precisa ser expressa na mesma representação intermediária à qual os documentos são submetidos, com o fim de garantir que as interfaces de documentos e requisições sejam compatíveis. Porém, a representação das requisições, diferentemente do que ocorre com os documentos, limita-se a identificar os termos que as compõem e não implica na materialização de um índice. A representação intermediária assim obtida é denominada consulta e serve à interrogação do espaço informacional do sistema de RI.

Tal interrogação processa-se por meio de algoritmos de casamento (*matching*), cujo comportamento é dependente das representações intermediárias adotadas nos documentos e nas requisições, bem como das características do modelo empregado. O resultado desses algoritmos de casamento é o atendimento da necessidade informacional, e sua forma é altamente dependente das particularidades do processo adotado. Por exemplo, os resultados podem se expressos por uma lista ordenada ou não-ordenada de documentos relevantes, por uma sumarização dos conteúdos relevantes, por classes de tópicos, por um esquema de visualização de informação, entre outros.

Por fim, a integração de todo o processo, a saber, as representações de documentos e requisições, a estratégia de casamento entre os mesmos, e a forma como os resultados são obtidos é abstraída em um arcabouço teórico denominado modelo de recuperação de informação. A seguir serão abordados os modelos de RI.

2.2 Modelos de Recuperação de Informação

Um modelo de recuperação de informação é o arcabouço teórico de um processo de recuperação de informação. O modelo é responsável por prover, ao processo, especificações e métodos para as representações intermediárias, tanto dos documentos,

quanto das requisições de informação, assim como estratégias para o casamento entre essas representações [Ingwersen and Järvelin 2005].

Ao longo do histórico das pesquisas em Recuperação de Informação, uma infinidade de modelos foram propostos. Tais modelos, além de numerosos, são classificados segundo critérios multidimensionais. Possíveis dimensões a considerar seriam a fundamentação matemática (e.g., baseados em teoria dos conjuntos, algébricos, probabilísticos), o grau de dependência entre os termos dos documentos (e.g. com dependência, sem dependência), a estratégia de casamento requisições-documentos (e.g. exata, parcial, difusa), o tratamento do espaço informacional dos documentos (e.g. topológicos, baseados em redes, baseados em árvores, baseados em hipertexto) [Baeza-Yates and Ribeiro-Neto 1999, Ingwersen and Järvelin 2005, Chowdhury 2003]. Dentre a infinidade de modelos propostos, podem-se citar entre alguns dos principais [Baeza-Yates and Ribeiro-Neto 1999, Chakrabarti 2002]:

Booleano: modelo de casamento exato baseado na teoria dos conjuntos. Os documentos são indexados considerando a premissa de independência entre os termos. Consultas são formuladas com a aplicação de operadores booleanos sobre os termos e processadas sobre o índice numa lógica similar à empregada em SGBDs relacionais;

Espaço de vetores: modelo de casamento parcial e fundamentação algébrica. Documentos e consultas são representados como vetores multidimensionais, com a premissa de independência entre os termos. O processamento da consulta dá-se por funções de similaridade algébrica entre os vetores [Salton et al. 1975];

Análise de Semântica Latente: modelo de casamento parcial e fundamentação algébrica que representa documentos e consultas numa matriz esparsa de correspondência, considerando dependências entre os termos. A indexação é conceitual, e os conceitos são definidos pela análise de relacionamentos entre os termos, por meio de decomposições sobre a matriz de correspondência [Foltz 1996];

Okapi BM25: modelo de casamento parcial e fundamentação probabilística, com premissa de independência de termos. O casamento é baseado na probabilidade de um documento ser relevante para uma consulta [Büttcher et al. 2006];

Modelagem de linguagem: modelo baseado em processamento de linguagem natural, em que documentos e consultas são representados por seqüências *n-gram*,

com o intuito de estabelecer uma distribuição probabilística entre os termos do *corpus*. O casamento ordena os documentos de acordo com a probabilidade do modelo de linguagem do documento gerar os termos da consulta [Kraaij 2005].

Aquém dos critérios de classificação, um modelo de recuperação de informação tem o fim de apoiar o usuário em ações especializadas denominadas tarefas de busca (*search tasks*). Uma tarefa de busca é uma ação, ativa ou pró-ativa, que envolve o usuário perante um sistema de recuperação de informação, visando a satisfação de uma necessidade informacional [Baeza-Yates and Ribeiro-Neto 1999]. Ativas são as tarefas de busca em que os documentos mantém-se relativamente estáticos enquanto são submetidos a um fluxo de requisições; pró-ativas são as tarefas de busca em que as consultas mantém-se relativamente estáticas enquanto são submetidas a um fluxo de documentos.

Dependendo de suas características, um modelo de recuperação de informação pode combinar diferentes tarefas de busca num mesmo sistema. Dentre as principais tarefas de busca encontradas na literatura, podem-se citar [Borlund 2003b, Ingwersen 1992a, Crestani and Ruthven 2007]:

Recuperação *ad-hoc*. Tarefa ativa, em que o usuário interage com o sistema com o intuito de obter indicadores de informação (ponteiros para documentos) que subsidiem a satisfação de sua necessidade informacional. Com isso, a necessidade só é considerada efetivamente satisfeita depois que o usuário inspecionar os documentos indicados e constatar que seus conteúdos a atendem.

Filtragem. Tarefa pró-ativa, em que perfis de consulta, criados pelos usuários, são mantidos pelo sistema. Cada novo documento é redirecionado aos usuários cujos perfis casam com o conteúdo do documento [Hanani et al. 2001].

Recomendação. Tarefa pró-ativa, com comportamento similar à filtragem. Porém, na recomendação, os perfis são obtidos pela análise dos padrões de consumo de informação do usuário, obtidos por monitoramento e/ou mineração de registros históricos de interação com o sistema [Adomavicius and Tuzhilin 2005]. Recomendação pode ser vista como a forma mais comum de personalização em recuperação de informações.

Navegação. Emprega técnicas de classificação com o intuito de dispor os documentos em estruturas (e.g. hierarquia, rede, nuvem) navegáveis e interativas que representam o espaço informacional do sistema [Mcdonald and Chen 2006].

Resposta a perguntas (*Question Answering*). Tarefa ativa que provê respostas de alto nível a consultas (perguntas) em linguagem natural. O mecanismo de resposta pode empregar técnicas de inteligência artificial, extração de informação, sumarização, análise de informações situacionais, entre outras [Agichtein et al. 2007].

Visualização de informação. Tarefa ativa ou pró-ativa, empregada na análise de grandes quantidades de documentos. Em geral é associada a uma tarefa de navegação, para permitir ao usuário explorar o espaço informacional enquanto o visualiza [Koshman 2006].

Dentre os diversos modelos propostos na literatura, um dos mais bem-sucedidos é o modelo de espaço de vetores (ou simplesmente modelo de vetores) [Salton et al. 1975], que prima pela simplicidade e pelo bom desempenho em coleções de propósito geral, mais comumente empregado em tarefas de busca *ad-hoc* e filtragem. Grande parte dos modelos propostos posteriormente ao modelo de vetores apresentam ganhos de desempenho que, ou são insuficientes para compensar suas abordagens mais complexas, ou apresentam ganhos que oscilam de acordo com as características das coleções de documentos.

Devido a essas vantagens, o modelo de espaço de vetores desfruta de grande popularidade na academia e na indústria. Em vista disso, e também pelo fato do modelo de vetores ser a base para os principais algoritmos de retroalimentação de relevância, a seguir será dedicada maior atenção a esse modelo, que é de particular interesse ao foco desta pesquisa.

Modelo de espaço de vetores

O princípio básico do modelo de espaço de vetores é a atribuição de pesos graduados nos termos das consultas e dos documentos. Os pesos assim atribuídos são usados para calcular o grau de similaridade entre os documentos armazenados no repositório e as consultas dos usuários. Com isso, torna-se possível efetuar casamento parcial e os documentos recuperados podem então ser ordenados segundo o grau de similaridade que apresentam com relação à consulta, introduzindo a noção de *ranking* de relevância.

Em situações reais, a maioria dos termos dos documentos tem conteúdo com baixo valor informacional (e.g., artigos, preposições, conjunções, verbos muito comuns). De forma a compactar o vocabulário, é comum pré-processar as representações, visando à

eliminação desses termos. Um possível pré-processamento com esse fim é apresentado na figura 2.2.

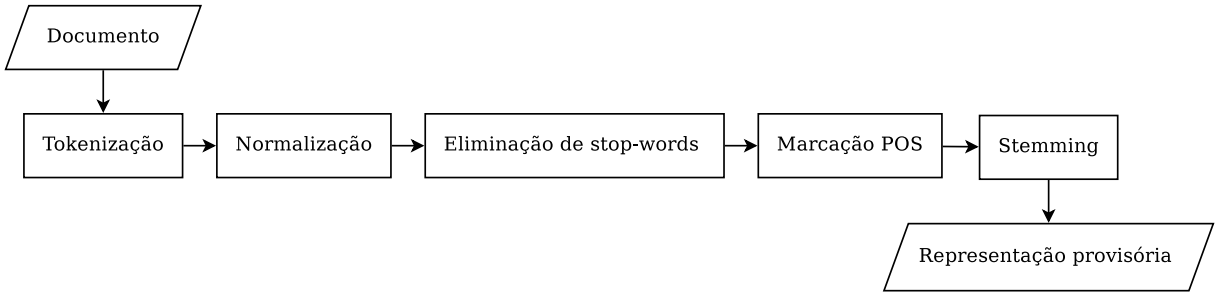

Figura 2.2: *Pipeline* para pré-processamento de documentos.

No *pipeline* da figura 2.2, cada estágio diminui o conjunto de termos da representação dos documentos, tornando-a mais compacta. O primeiro estágio de processamento apresentado realiza a decomposição do texto em um conjunto de fragmentos, denominados *tokens*, excluindo-se sinais de pontuação. Os tokens sofrem então uma normalização para garantir, por exemplo, que todos estejam em caixa baixa. As palavras com baixo valor informacional, denominadas *stop-words*, são subtraídas do conjunto de *tokens*. O próximo estágio, marcação POS (*part of speech* ou parte de discurso), identifica a classe gramatical do token; possuindo essa classificação, o sistema pode manter apenas os substantivos, por exemplo. O estágio de *stemming* realiza a eliminação de prefixos e sufixos dos tokens. Após a execução de todos os estágios, o vocabulário do documento estará mais compacto, o que otimizará o processamento da coleção e diminuirá a ocorrência de ruídos no processo de casamento.

As representações provisórias obtidas pelas tarefas de pré-processamento são indexadas pelo modelo de espaço de vetores, que pode ser definido formalmente por uma quádrupla $\langle D, Q, F, R(q_i, d_j)\rangle$, onde:

1. D é o conjunto formado pelas representações de documentos da coleção;
2. Q é o conjunto formado pelas representações de requisições (consultas) emitidas ao sistema;
3. F é um *framework* que sentencia como se dão as representações de documentos e requisições, assim como o casamento entre elas;
4. $R(q_i, d_j)$ é uma função de ordenação que atribui um número real à associação entre a consulta q_i e o documento d_j.

A principal definição proferida pelo *framework F* remete à representação dos documentos e requisições. Para tal, sejam k_i uma chave de indexação, d_j uma representação de documento e $w_{i,j} \in [0,1]$ um peso associado ao par (k_i, d_j). Cada peso $w_{i,j}$ é um valor não-binário que pode ser determinado por alguma métrica estatística e corresponde à importância do termo k_i com relação ao conteúdo do documento d_j. Sejam t a quantidade de termos (chaves de indexação) em D (isto é, o vocabulário), k_i um termo arbitrário do vocabulário e $K = \{k_1, k_2, ..., k_t\}$ o conjunto de todos os termos do vocabulário. Um peso $w_{i,j} \in (0,1]$ é associado a cada par (k_i, d_j), desde que k_i esteja presente em d_j; se k_i não está presente em d_j, então $w_{i,j} = 0$.

Para representar o vocabulário dos documentos, associa-se a cada d_j um vetor $\vec{d}_j$ tal que $\vec{d}_j = \langle w_{1,j}, w_{2,j}, ..., w_{t,j} \rangle$. Com isso, cada documento da coleção é representado como um vetor cuja dimensão é igual à cardinalidade do vocabulário da coleção. A mesma estratégia de representação é adotada para as requisições, de forma a garantir que representações de documentos e requisições mantenham interfaces compatíveis. Logo, para uma consulta $q \in Q$ associa-se um vetor $\vec{q}$ tal que $\vec{q} = \langle w_{1,q}, w_{2,q}, ..., w_{t,q} \rangle$. Conseqüentemente, tanto documentos quanto requisições podem ser dispostos num espaço t-dimensional, como ilustrado na figura 2.3.

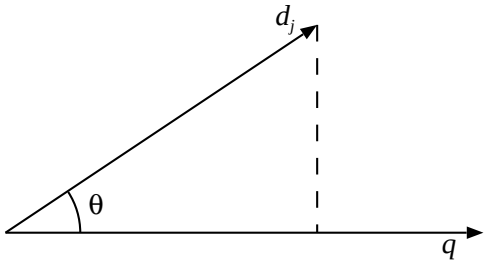

Figura 2.3: Documento d_j e consulta q representados como vetores num espaço t-dimensional, com $t = 2$.

Para obter a função de ordenação R prevista no modelo, é necessário calcular o grau de similaridade entre requisições e documentos, usando a correlação entre os vetores que os representam, $\vec{q}$ e $\vec{d}_j$, respectivamente. Uma possível estratégia para obter o grau de similaridade, por essa correlação, é calcular o cosseno do ângulo θ entre os dois vetores:

$$R(d_j, q) = \cos\theta = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}}$$

O conjunto dos valores $R(d_j, q)$ representa o casamento parcial da consulta q com a coleção D. O caráter parcial desse processo permite estabelecer uma ordenação, ou

ranking, dos documentos em relação à consulta, no qual os documentos mais relevantes estarão no topo do *ranking* de resultados.

Resta agora definir uma métrica para atribuir pesos aos termos. Dentre as várias abordagens com esse intuito, uma das mais efetivas é a métrica *tf-idf* e suas extensões [Salton and Buckley 1988]. A medida *tf* (*term frequency*) tem escopo local e indica qual a importância de um termo para um documento em específico. Por outro lado, a medida *idf* (*inverse document frequency*) tem escopo global e indica a distribuição do mesmo termo na coleção como um todo.

A forma básica dessa métrica é definida como segue: seja $freq_{i,j}$ o número de vezes que o termo k_i aparece no documento d_j, ou seja, a freqüência bruta. Define-se $tf_{i,j}$ como a freqüência normalizada do termo k_i no documento d_j, que é dada por:

$$tf_{i,j} = \frac{freq_{i,j}}{max_l freq_{l,j}}$$

Onde $max_l freq_{l,j}$ é a maior freqüência computada entre os termos de d_j, aplicada para efetuar a normalização da freqüência bruta.

Define-se também df_i como a freqüência bruta do termo k_i em toda a coleção. Seja N o número de documentos da coleção e n_i o número de documentos que contém o termo k_i. Para que o peso do termo k_i seja inversamente proporcional a df_i, define-se idf_i como a freqüência inversa e normalizada do termo k_i em todos os documentos da coleção:

$$idf_i = \log \frac{N}{n_i}$$

Aplicando $tf_{i,j}$ e idf_i em conjunto para a definição dos pesos, é possível quantificar a importância de um termo para um documento, considerando a importância do mesmo termo para a coleção: $tf_{i,j}$ mensura o poder de descrição do termo, para o documento; já idf_i mensura o poder de discriminação do mesmo termo, na coleção. Os pesos podem ser então definidos como o produto entre as duas medidas:

$w_{i,j} = tf_{i,j} \times idf_i$, para documentos, e $w_{i,q} = tf_{i,q} \times idf_i$, para consultas.

Expostos os principais modelos de RI e concedida ênfase ao modelo de espaço de vetores, a seção 2.3 trata de metodologias para avaliar sistemas de RI.

2.3 Avaliação de sistemas de recuperação de informação

O objetivo central de um sistema de recuperação de informação é maximizar a recuperação de documentos relevantes e minimizar a recuperação de documentos irrelevantes, em vista das necessidades informacionais investigadas no sistema. Nota-se, com isso, que a noção de relevância é fundamental para se avaliar um sistema de RI. Em respeito a essa característica, duas vertentes, cada uma preocupando-se com conceituações e aspectos distintos de relevância, merecem destaque neste trabalho: a avaliação centrada no sistema (seção 2.3.1), voltada a fatores objetivos, e a avaliação centrada no usuário (seção 2.3.2), atenta a fatores objetivos e subjetivos.

2.3.1 Avaliação centrada no sistema

A avaliação centrada no sistema, também denominada de abordagem *Cranfield* [Baeza-Yates and Ribeiro-Neto 1999], é a mais tradicional e dominante na literatura de RI. Por essa abordagem, os experimentos de avaliação são executados, em lote, com variáveis controladas, isto é, desconsiderando a influência de fatores humanos. Em face a essas características, os experimentos centrados no sistema desfrutam das vantagens de serem facilmente reproduzíveis e escaláveis, facilitando a comparação quantitativa entre diferentes implementações. Para viabilizar comparações, é convencional o emprego de coleções de teste padronizadas (e.g. TREC [Voorhees 2005], Cranfield, INEX [Kazai et al. 2003]), compostas por três recursos:

1. um *corpus* de documentos;
2. um conjunto de requisições de informação; e
3. um conjunto de julgamentos de relevância, mapeando documentos e consultas.

A figura 2.4 estende a figura 2.1 para ilustrar como as tarefas de avaliação integram-se às tarefas típicas de um processo de RI; nessa figura, as tarefas representadas como formas sombreadas são as que sofrem intervenção durante a avaliação.

As requisições são formuladas por especialistas no domínio dos documentos, e em geral constituem-se da expressão textual de uma necessidade informacional e uma consulta que traduza essa necessidade. Alternativamente, as consultas podem ser coletadas de logs de sistemas legados que tenham previamente atuado sobre o mesmo *corpus*.

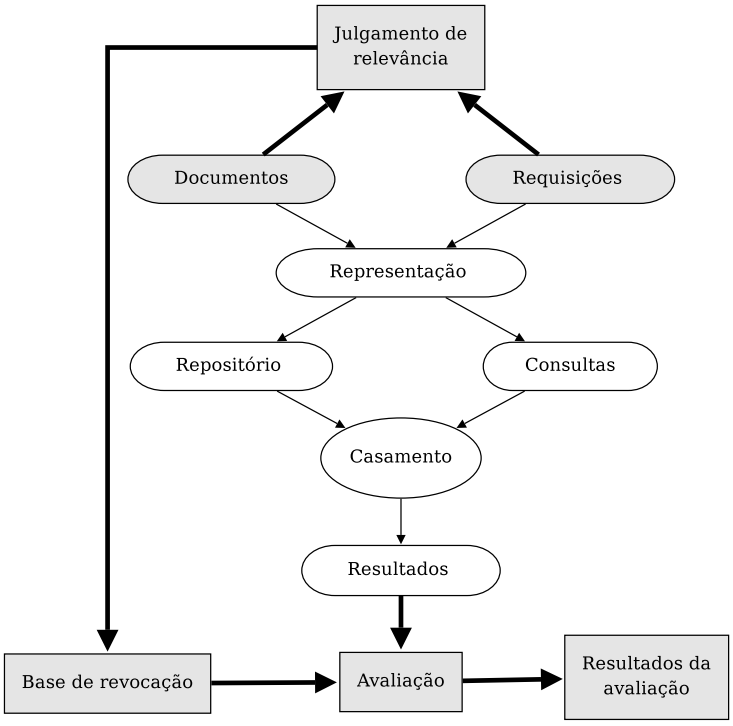

Figura 2.4: Tarefas da metodologia de laboratório para avaliação de sistemas de recuperação de informação. Adaptado a partir de [Ingwersen and Järvelin 2005].

Os julgamentos de relevância, por sua vez, estabelecem quais documentos são relevantes para cada requisição. Tais mapeamentos podem ser realizados por curadores humanos, especialistas nos tópicos tratados na coleção. Porém, para coleções de grande porte, em que o emprego de curadores humanos pode ser custoso e proibitivo, é comum adotar-se o método de *pooling* [Voorhees 2005], largamente usado na comunidade de RI, para obter os julgamentos de relevância. Este método constitui-se na submissão de uma requisição a vários sistemas de bom desempenho, obtendo em resposta um conjunto de resultados para cada sistema. Os melhores resultados assim obtidos são então acumulados em um *pool* que é intelectualmente inspecionado para formar os julgamentos de relevância para a requisição.

Embora a expressão mais comum de mapeamento sejam julgamentos binários (relevante ou não-relevante), abordagens mais recentes empregam julgamentos graduados [Xu and Chen 2006, Mechmache et al.]. O conjunto formado por todos os mapeamentos materializa-se em uma base de revocação, cuja finalidade é servir de referência ao cômputo das métricas de desempenho. Para executar um experimento, submetem-se as consultas pré-formuladas ao sistema, em lote e, para os resultados de cada consulta, computam-se métricas a partir da base de revocação. Por fim, os dados assim obtidos são analisados, plotados ou aproveitados em métricas complementares que os

sumarizem.

As métricas clássicas de desempenho em RI são a **precisão** (*precision*) e a **revocação** (*recall*). Definindo-as formalmente, sejam q a representação de uma requisição de informação proveniente de uma coleção de teste e R o conjunto de documentos julgados como relevantes para q. Seja $|R|$ o número de documentos em R. Ao emitir q no sistema sob avaliação, obtém-se um conjunto de resultados A, com $|A|$ documentos. Por fim, seja $|R_a|$ o número de documentos presentes na intersecção entre R e A. A figura 2.5 apresenta graficamente a relação entre os conjuntos R, A e R_a.

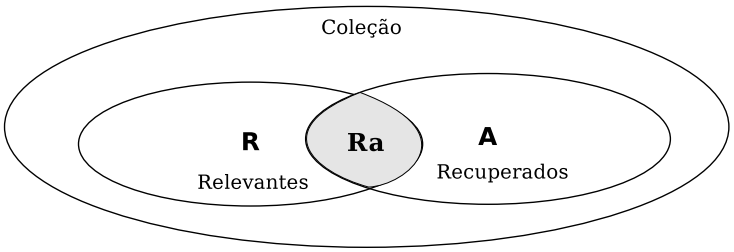

Figura 2.5: Relação entre conjuntos de documentos considerados pela precisão e pela revocação.

Revocação é definida como a fração dos documentos relevantes que foi recuperada, ou melhor:

$$revocação = \frac{|R \cap A|}{|A|} = \frac{|R_a|}{|A|}$$

Precisão é definida como a fração dos documentos recuperados que é julgada como relevante, ou melhor:

$$precisão = \frac{|R \cap A|}{|R|} = \frac{|R_a|}{|R|}$$

Observando essas métricas, é possível afirmar que um sistema hipotético, que sempre recupere todos os documentos da coleção, para qualquer consulta, terá sempre revocação máxima. Ou ainda, que um sistema que sempre apresente apenas o resultado com o maior grau de similaridade, tenderá a maximizar a precisão. Para compensar essas degenerações, precisão e revocação são sempre estudadas simultaneamente para avaliar o sistema.

Em geral, um sistema considerado como eficiente é capaz de apresentar um bom equilíbrio entre precisão e revocação, embora esse pressuposto possa não se aplicar dependendo dos requisitos considerados. Por exemplo, mecanismos de buscas da Web, dada a grande quantidade de resultados que apresentam, privilegiam que os resultados na primeira página sejam os mais relevantes, favorecendo a revocação. Por outro

lado, em algumas coleções de domínio específico, a exemplo das bases de legislação e de serviços de inteligência, é comum que todos os resultados relevantes sejam inspecionados, favorecendo a precisão [Chakrabarti 2002].

Precisão e revocação são os blocos fundamentais das métricas de avaliação formal dos algoritmos para recuperação de informação. A partir delas, podem-se promover diferentes análises dos resultados, como plotagem dos seus valores em gráficos de precisão *vs.* revocação, interpolação dos valores com restrição no limiar de pontos, histogramas, médias que sumarizam as duas métricas, entre outras. Como os resultados serão formalmente apresentados, e como serão comparados com os de outros sistemas, é uma decisão fortemente influenciada pelos requisitos dos sistemas avaliados ou pelas diretrizes das conferências especializadas em avaliação (e.g. TREC, INEX), que se tomem como referência.

Uma forma bastante difundida de apresentar os resultados de um sistema de RI, empregada pela comunidade participante da conferência TREC, é o gráfico de precisão média em 11 pontos de revocação. Nesta representação, a precisão média para cada requisição de informação é calculada tomando como referência 11 pontos discretos no intervalo $[0, 1]$, representando os possíveis níveis de revocação. Este tipo de sumarização dos dados pode ser facilmente obtida pela ferramenta *trec_eval*[1], disponibilizada livremente pela conferência TREC. A precisão média privilegia o fato de que documentos no topo da lista de resultados devem ter importância maior. A figura 2.6 apresenta um exemplo de gráfico de precisão média em 11 pontos de revocação.

A partir da figura 2.6, é possível notar 11 níveis de revocação (0.1, 0.2, 0.3, 0.4, ..., 1) em que a precisão média é calculada. A precisão média no nível 0.2, constitui a precisão aferida nos resultados após a inspeção de 20% dos resultados a partir do topo; em 0.4, a precisão média após inspecionar 40% dos resultados; e assim por diante. A partir desse gráfico é possível visualizar em que momentos o sistema apresenta melhor desempenho. Por exemplo, se o sistema apresenta alta precisão nos primeiros níveis de revocação, então a densidade de documentos no topo do *ranking* é alta. Deste modo, é possível comparar o desempenho de dois sistemas comparando suas precisões médias em cada nível de revocação.

Embora a avaliação centrada no sistema desfrute da vantagem de viabilizar experimentos restritos ao ambiente controlado dos laboratórios, ao mesmo tempo é alvo de fortes críticas devido à sua ênfase na mera avaliação quantitativa dos algoritmos

[1] `http://trec.nist.gov/trec_eval/`

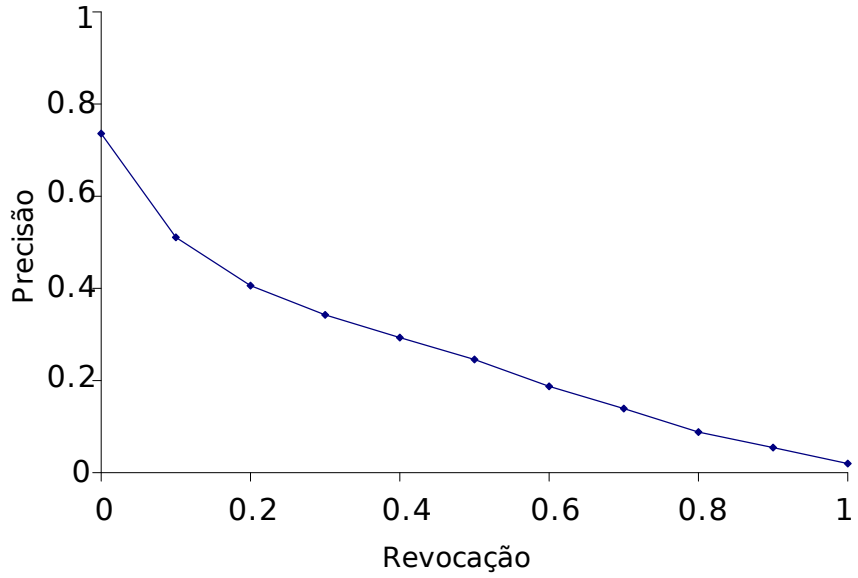

Figura 2.6: Exemplo de gráfico para precisão média interpolada em 11 níveis de revocação. Adaptado de [Manning et al. 2008]

para recuperação de informação. Para suplantar essas limitações, algumas abordagens propõem-se a inserir sujeitos humanos no experimento de avaliação, caracterizando-se como abordagens que consideram aspectos interativos [Borlund 2003b] e cognitivos [Ingwersen and Järvelin 2005] da noção de relevância. Tais abordagens denominam-se avaliações centradas no usuário e serão tratadas na próxima seção.

2.3.2 Avaliação centrada no usuário

As abordagens de avaliação centradas no usuário, inspiradas nos conceitos de Interação Humano-Computador (IHC), estendem o modelo tradicional de avaliação — centrado no sistema — para viabilizar um ambiente propício à análise de critérios objetivos e subjetivos da relevância, bem como propõem novas métricas para estimá-la [Wilkinson and Wu 2004]. A figura 2.7 ilustra modificações introduzidas pelas metodologias orientadas ao usuário na avaliação de sistemas de recuperação de informação.

As metodologias orientados ao usuário suprimem as consultas pré-definidas e os julgamentos de relevância obtidos de especialistas. Para substituir esses recursos, seleciona-se etnograficamente um grupo de sujeitos humanos, aos quais são fornecidas histórias curtas, que expressam necessidades informacionais reais. Tais histórias, denominadas situações de tarefas simuladas (figura 2.8), descrevem: o disparador da necessidade informacional; o ambiente em que a situação ocorre; o objetivo da busca por informa-

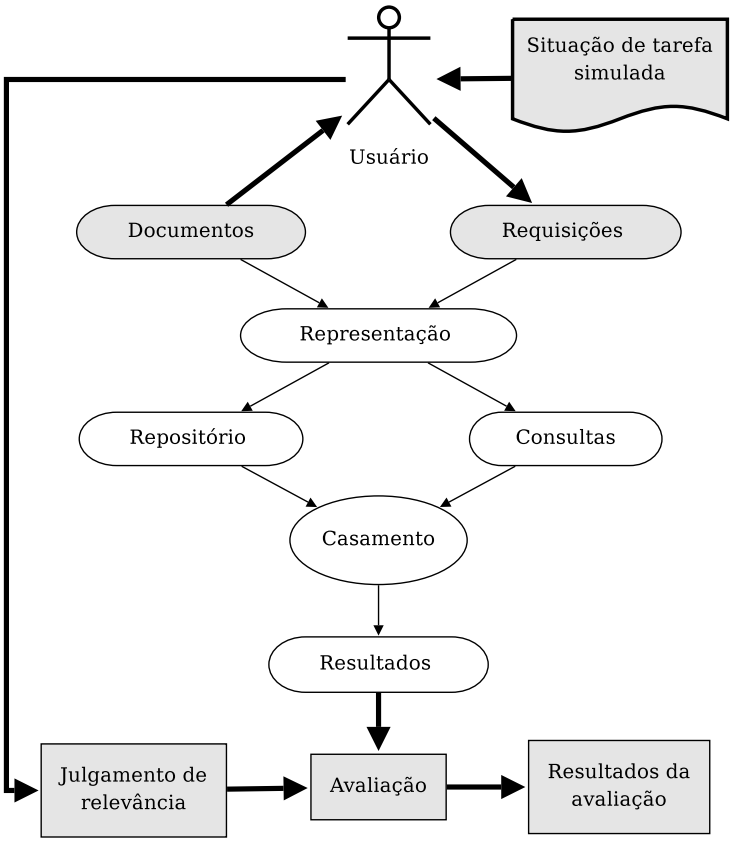

Figura 2.7: Contribuições das metodologias orientadas ao usuário para avaliação de sistemas de recuperação de informação.

ção; e, opcionalmente, um indicador de como a requisição de informação poderia ser formulada.

Situação simulada

Situação de tarefa simulada: após sua graduação, você irá pleitear um emprego na indústria. Para tal, você precisa de informações que o auxiliem a focalizar sua procura por trabalho, pois sabe que é difícil conhecer o mercado. Você gostaria de encontrar informações sobre padrões de trabalho na indústria e que tipo de qualificações os empregadores desejam de seus futuros empregados.

Requisição indicativa: Procure, por exemplo, algo sobre projeções de empregabilidade na indústria, isto é, áreas que apresentem expansão ou retração de postos de trabalho.

Figura 2.8: Exemplo de uma situação de tarefa simulada.

A requisição de informação, formulada pelo usuário, a partir da situação de tarefa simulada, é emitida ao sistema. Obtidos os resultados, o usuário intelectualmente julga quais documentos são relevantes para suas consultas, aplicando valores de uma escala. Os julgamentos de relevância assim obtidos apóiam a computação de um conjunto de métricas quantitativas, que incluem tanto as empregadas na abordagem tradicional, quanto métricas alternativas e menos difundidas que consideram acessibilidade, usa-

bilidade, desempenho do usuário, entre outros fatores. Opcionalmente, podem ser empregados métodos adicionais de coleta de dados, como entrevistas, questionários no início e no fim do experimento, observações, monitoramento de ações e gravações em vídeo [Borlund 2003a].

Embora ofereçam um tratamento mais completo da relevância, as metodologias centradas no usuário são experimentos caros, de projeto complexo e difíceis de reproduzir [Goker and Myrhaug 2008]. Em geral, os resultados experimentais são bastante dependentes da usabilidade da interface do sistema, da experiência do sujeito com aplicações de RI e do grau de compreensão das tarefas simuladas pelo sujeito, entre outras interferências subjetivas.

Dados esses dificultadores, em sistemas nos quais os custos da configuração particular de um experimento centrado no usuário são proibitivos, mas nos quais recursos interativos são cruciais, é comum o uso de alternativas mais leves — porém menos completas — de estudar o usuário.

Uma destas abordagens é o monitoramento de cliques (*clickthrough behavior*), que consiste no registro dos documentos selecionados pelo usuário na lista de resultados, e posterior análise para derivar os julgamentos de relevância. Esse método desfruta da vantagem de eliminar a necessidade de julgamentos explícitos, apoiando-se em sinais implícitos que podem ser obtidos a baixo custo e sem sobrecarga cognitiva do usuário. Nos experimentos realizados em [Joachims 2002], nota-se que os julgamentos assim obtidos tem qualidade próxima aos julgamentos obtidos explicitamente. Resultados semelhantes foram observados em [Jung et al. 2007b] e [Shen et al. 2005b].

Uma abordagem menos comum, denominada simulações de usuário, consiste na modelagem de agentes que comportem-se como usuários de um sistema de RI. Para tal, os agentes são submetidos a aprendizado a partir de logs de interação ou consultas de usuários reais. Tais informações podem ser obtidas instrumentando-se o sistema e submetendo os registros resultantes a tarefas analíticas, a exemplo de mineração de dados [Whittle et al. 2007]. Em [Lin 2007] essa abordagem é empregada em sistemas de respostas a perguntas e em [White et al. 2004] é aplicado para avaliar estratégias de expansão de consulta.

2.4 Considerações finais

Este capítulo introduziu o tema de Recuperação de Informação, enfatizando seus processos, modelos e metodologias de avaliação. Nesse sentido, foram expostas as tarefas típicas de um processo de RI, e como essas tarefas relacionam-se com modelos; em particular, enfatizou-se o modelo de espaço de vetores, de especial interesse a essa monografia. Por fim, trataram-se as duas principais vertentes de avaliação de sistemas de RI: a avaliação centrada no sistema e a avaliação centrada no usuário.

O próximo capítulo abordará como o contexto das necessidades informacionais influencia os sistemas de RI. Serão apresentadas as principais tendências de pesquisa, com ênfase nas de contexto relacionadas à Computação Ubíqua, à Personalização e à Recuperação Interativa de Informação.

3 *Contexto em Recuperação de Informação*

O conceito de *relevância* é um princípio que fundamenta tanto o funcionamento quanto a avaliação de sistemas de RI, visto que o principal objetivo desses sistemas é a seleção de documentos que sejam relevantes à necessidade informacional do usuário. Entendimentos recentes definem relevância como um fenômeno inerentemente dinâmico e multidimensional [Cosijn and Ingwersen 2000, Borlund 2003a], pois é dependente da satisfação direta das necessidades informacionais de usuários reais. O caráter dinâmico desse conceito diz respeito às variações, ao longo do tempo, da forma como um mesmo usuário percebe o espaço informacional com o qual está interagindo. Já o caráter multidimensional da relevância baseia-se no fato de que diferentes sujeitos humanos percebem e atestam documentos relevantes de formas distintas, evidenciando que múltiplas dimensões de relevância devem ser consideradas como variáveis para avaliar esse fenômeno.

Dentre essas dimensões, a que se mostra mais fiel às necessidades informacionais de usuários reais é a que considera evidências situacionais [Xu and Chen 2006]. Relevância situacional é vista como a utilidade dos objetos informacionais com relação à tarefa de trabalho em que o usuário está envolvido no momento em que surge a necessidade informacional, ou seja, a situação do usuário no momento da busca. Sendo assim, esse tipo de relevância é altamente dependente do *contexto* em que o usuário está inserido e é potencialmente dinâmica.

A partir de uma interpretação mais prática da influência do contexto no conceito de relevância, é possível identificar dois parâmetros que caracterizam uma necessidade informacional [Hernandez et al. 2007]: o tema, expresso explicitamente pela requisição de informação (termos de busca); e o contexto, que denota a causa da busca da informação e com qual finalidade ela será usada. Embora o tema da necessidade informacional seja um aspecto bastante consolidado, o contexto apresenta expressiva hete-

rogeneidade de definições e é apontado como um dos grandes desafios em pesquisas de RI [Allan et al. 2003].

A habilidade de responder ao contexto viabiliza aos sistemas de RI aprender e predizer qual informação os usuários necessitam, decidir como e quando tal informação precisa ser apresentada e distinguir entre diferentes tipos de tarefas de busca e preferências de usuários [Crestani and Ruthven 2007]. Com isso, decisões importantes de sistemas que trabalham com contexto consistem em definir quais aspectos de contexto devem ser considerados, como as evidências contextuais podem integrar-se aos modelos de RI e como integrar fontes potencialmente heterogêneas de contexto.

É evidente a coexistência de várias definições de contexto em uso atualmente, cada uma refletindo as particularidades da área de pesquisa do autor que a propôs. Com efeito, em [Bazire and Brézillon 2005] foram coletadas por volta de 150 diferentes definições de contexto, oriundas de diversas disciplinas, com o fim de analisá-las semanticamente. Mesmo quando situadas as interpretações presentes em pesquisas de RI, em específico, nota-se uma ampla diversidade de interpretações sobre o conceito de contexto, implicando também em não menos diverso número de abordagens e aplicações: proposição de modelos, estratégias de indexação, esquemas de apresentação de resultados, metáforas de interação com resultados, entre outras.

Em vista de tal multiplicidade de interpretações de contexto e consequente amplitude de abordagens, ressalta-se que este capítulo limita seu escopo especificamente às pesquisas que aproveitam evidências de contexto em cenários de recuperação de informação, com ênfase em modificação de consulta. De modo a fornecer um painel geral das principais pesquisas sob esse escopo, pode-se distinguir as seguintes tendências:

Vizinhança de termos como contexto: principal abordagem para lidar com contexto em sistemas de RI, consiste em minerar, nos objetos informacionais da coleção, freqüência, localização e co-ocorrência entre seus termos; trata-se de um tipo de contexto que remete aos aspectos lingüísticos da informação, de forma a identificar seqüências de termos semanticamente similares. O resultado dessas análises pode ser então usado em diversas técnicas automáticas, como retroalimentação cega de relevância e indexação de semântica latente. Abordagens mais recentes exploram o contexto lingüístico em cenários interativos, permitindo ao usuário emitir requisições baseadas em trechos de documentos.

Domínio como contexto: alguns sistemas de RI que servem a domínios específicos,

exploram a homogeneidade contextual do domínio para reduzir a ambigüidade lingüística de documentos e consultas. Em geral, tal feito é obtido com a aplicação de vocabulários controlados durante a expansão de consultas, revelando-se como uma interpretação de contexto fundamentalmente estática e limitada.

Ambiente como contexto: muito comum em sistemas de RI para dispositivos móveis e Computação Ubíqua. Nesse caso, contexto remete à premissa de que a dinamicidade da relevância é influenciada por mudanças em variáveis do ambiente físico em que o usuário está inserido, bem como por características do meio de acesso empregado. Dessa forma, é possível reformular consultas visando privilegiar os resultados mais adequados a configurações ambientais específicas.

Personalização como contexto: personalização ocorre em RI quando o sistema acumula um histórico de consultas e documentos com os quais o usuário interagiu e o aplica para refinar recuperações futuras. Isso envolve modelagem, aprendizado e coordenação de modelos de usuário, abrangendo preferências de curto ou longo prazos, isto é, preferências do usuário ao longo do tempo. Dessa forma, dois usuários com a mesma consulta deparam-se com diferentes resultados pois os contextos apreendidos pelo sistema diferem para os dois sujeitos. Outros avanços interpretam o contexto como as interações sociais em comunidades de usuários, sob abordagens de filtragem colaborativa e RI para redes sociais.

Espaço informacional como contexto: em algumas coleções cujos documentos estabelecem referências entre si, a exemplo dos corpora de hipertexto e de artigos científicos, o conteúdo e a estrutura do espaço (ou hiperespaço, no caso de hipertexto) informacional que envolve os documentos é tido como um importante fator contextualizador. Dessa forma, as interações entre os objetos informacionais podem ser mensuradas, via técnicas bibliométricas ou *web*-ométricas, para atribuir graus de autoridade aos documentos e assim melhorar o *ranking* de resultados.

Tarefas como contexto: considera as tarefas de trabalho, geradoras das tarefas de busca, como contextualizadores. As tarefas de trabalho permeiam o contexto cognitivo com o qual o usuário está lidando durante a tarefa de busca. Em geral, definem-se perfis de acesso, similarmente aos empregados em personalização, para identificar termos que ocorrem seguidamente em requisições e remetem à mesma tarefa de trabalho. Com esses subsídios, o sistema pode, por exemplo, usufruir de uma classificação de tarefas — estática ou dinâmica — e detectar qual dentre elas está

em evidência durante a interação, privilegiando a recuperação dos documentos mais relevantes na recorrência de uma dada situação.

Como resultado de qualquer esforço de classificação de abordagens de pesquisa, muitas vezes nota-se que as fronteiras de certas abordagens submetidas à classificação são difusas e movediças. Por exemplo, é comum encontrar na literatura abordagens que advocam tratar de personalização e que ao mesmo tempo adotam a tendência de representar contexto como vizinhança de termos.

Na abordagem tratada nesta dissertação ocorre a influência convergente de três tendências: ambiente como contexto (segundo o ponto de vista da Computação Ubíqua); personalização como contexto; e tarefas como contexto (segundo o ponto de vista de Recuperação Interativa de Informação). Desse modo, no restante deste capítulo, será dada ênfase a essas tendências em específico. Na seção 3.1 será abordado o papel do contexto na Computação Ubíqua; na seção 3.3 será visto como o contexto influencia a personalização do acesso à informação; e a seção 3.2 discorrerá sobre o papel do contexto no âmbito da recuperação interativa de informação.

3.1 Contexto na Computação Ubíqua

A Computação Ubíqua é uma visão que surgiu nos laboratórios PARC da Xerox no final de década de 1980. Mark Weiser, autor do trabalho que inaugurou essa visão [Weiser 1999a], sentencia que a computação ubíqua objetiva à otimização do uso de computadores, tornando-os disponíveis no ambiente físico ao mesmo tempo que os torna invisíveis ao usuário. Trata-se da visão de um mundo no qual o custo do poder computacional e das comunicações digitais torna-se tão barato a ponto de se poder embutí-los em todos os objetos que nos cercam no dia-a-dia [Stajano 2002], ou seja, a onipresença do computador.

Nesse sentido acredita-se que as tecnologias mais profundas são aquelas que desaparecem — que tornam-se imperceptíveis. Elas se entrelaçam nas texturas do dia-a-dia da vida de seus usuários até tornarem-se indistinguíveis [Weiser 1999b], ou seja, são usadas sem serem notadas. Grandes avanços têm sido observados na direção de concretizar a visão de Weiser. Podem-se citar a concepção de computadores trajáveis para monitoramento de sinais vitais e hábitos humanos; casas sensíveis que aprendem os hábitos de seus moradores; carros sensíveis cujos sensores permitem interação com o mundo exterior para estender a interface tradicional do motorista; salas de aula ou de

reuniões, inteligentes, em que as telas e telões são capazes de operar autonomamente baseando-se no contexto percebido.

Concorrentemente a esses resultados outros desafios de pesquisa são enfrentados pela comunidade de pesquisa em Computação Ubíqua e, entre os que mais têm recebido atenção da comunidade científica, citam-se a necessidade de interfaces naturais; limitações dos recursos dos dispositivos de acesso; e sistemas sensíveis ao contexto em que estão inseridos.

Em específico, o estudo de sistemas sensíveis ao contexto tem sido bastante ativo e frequente, principalmente devido ao potencial de gerar aplicações inovadoras e personalizadas às necessidades dos usuários. Tais pesquisas tem identificado requisitos e proposto soluções concretas para diversos problemas relacionados à modelagem, à coleta e ao processamento de informações de contexto, bem como no projeto de aplicações. Nas próximas subseções, são dadas vistas a alguns desses aspectos: na seção 3.1.1 são discutidos sistemas sensíveis ao contexto e a seção 3.1.2 trata sobre modelos de contexto.

3.1.1 Sistemas sensíveis ao contexto

Sistemas sensíveis ao contexto utilizam informações de contexto para fornecer serviços ou informações relevantes a um usuário, sendo que a relevância está diretamente relacionada à tarefa desempenhada pelo usuário em um dado momento [Dey 2001].

Os primeiros sistemas sensíveis ao contexto eram cientes de localização. Porém, localização é apenas uma das dimensões em que as aplicações computacionais se contextualizam. Embora essa seja historicamente a dimensão mais popular, diversas pesquisas têm sido empreendidas para explorar outras dimensões. Posto que contexto excede localização (e, implicitamente à localização, também identidade) nota-se potencial para explorar outras dimensões como *tempo*, *história*, *sociedade*, entre as principais.

Mesmo tomando as pesquisas de computação ubíqua isoladamente, nota-se que há várias definições de contexto em uso atualmente. Em lugar de expô-las em grandes detalhes, adota-se nesta dissertação a definição mais amplamente aceita em computação ubíqua e ciência de contexto. Por essa definição, contexto é qualquer informação que pode ser usada para caracterizar a situação de uma entidade (e.g. atividade, identidade, localização, tempo, motivação); uma entidade pode ser qualquer coisa considerada relevante para a interação entre o usuário e a aplicação, incluindo os próprios

usuário e aplicação [Dey 2001]. De forma a operacionalizar essa definição, é amplamente aceito um conjunto mínimo de dimensões contextuais que um sistema sensível ao contexto deve considerar, denominado "5W" [Abowd and Mynatt 2000]:

- ***Who***: identidade do usuário e de outras pessoas relevantes ao usuário no ambiente;
- ***What***: interpretação das atividades que o usuário realiza;
- ***Where***: ciência dos movimentos do mundo físico;
- ***When***: compreensão do fluxo do tempo em apoio à interpretação das atividades do usuário;
- ***Why***: compreensão das finalidades das atividades que o usuário realiza.

A combinação dessas dimensões em modelos de contexto, viabiliza a construção de aplicações mais inteligentes, operantes sobre atributos oriundos do mundo real, e capazes de melhor interpretar as necessidades dos usuários. Em [Dey and Abowd 1999] são definidos cinco principais grupos de sistemas sensíveis ao contexto:

1. **Percepção de contexto**: consiste na coleta de dados contextuais, por meio de redes de sensores, por exemplo;
2. **Associação entre contexto e dados**: exemplificando, notas de reuniões podem ser associadas com as pessoas presentes e com o local da reunião;
3. **Descoberta contextual de serviços**: ativação de um periférico que esteja mais próximo do usuário, por exemplo;
4. **Ações disparadas por contexto**: cita-se como exemplo o carregamento de mapas quando se entra em uma determinada região;
5. **Mediação contextual**: como aplicação geral, cita-se a filtragem de um grande volume de dados para apresentação, baseando-se no que é interessante no atual contexto.

A partir de uma análise abrangente dos atuais sistemas sensíveis ao contexto, é possível notar alguns requisitos recorrentes. De forma geral, esses sistemas operam sobre

um conjunto de fontes de contexto potencialmente heterogêneas (usualmente sensores, logs de aplicações, perfis, entre outras), traduzindo os dados brutos dessas fontes em abstrações estabelecidas em modelos de contexto. Os modelos de contexto tendem a ser formados por representações de alto nível, com poder de expressão para descrever e integrar a variedade de dados presente no conjunto de fontes. Com isso, outro aspecto importante a ser tratado é a coleta e a interpretação dos dados de contexto.

Esses requisitos são recorrentes a inúmeras aplicações sensíveis ao contexto, logo é prática comum fatorá-los em infra-estruturas reusáveis e extensíveis — *frameworks*, *middlewares*, arquiteturas, etc — tal que possam ser compartilhadas por diversas aplicações. Uma possível generalização dessas infra-estruturas é apresentada na figura 3.1.

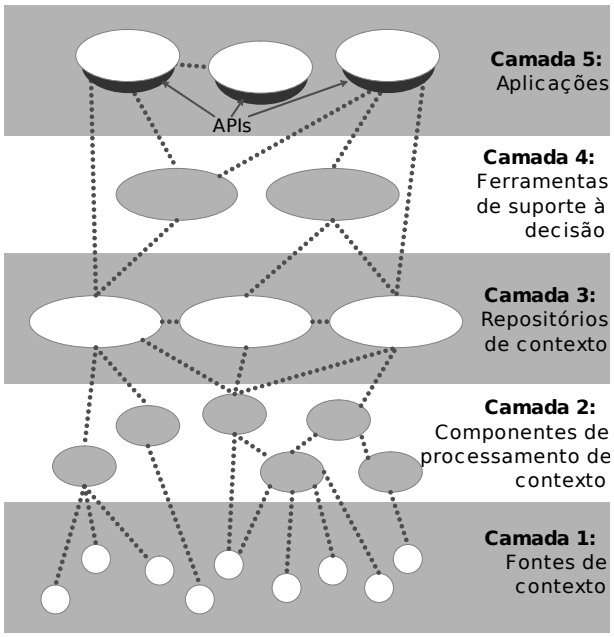

Figura 3.1: Arquitetura generalizada (ou arquitetura de referência) para sistemas sensíveis ao contexto (adaptado de [Henricksen et al. 2005])

De acordo com a figura 3.1, na camada 1 situam-se as fontes de contexto, que fornecerão dados brutos de natureza heterogênea. Na camada 2, esses dados brutos são coletados e interpretados, em referência ao modelo de contexto, por diversos adaptadores especializados. Já convertidos para instâncias do modelo de contexto, os dados são então armazenados em repositórios de contexto (camada 3). Esses repositórios serão mais tarde consultados, a partir da camada 4, por ferramentas de suporte a decisão (e.g. raciocionadores, modelos de aprendizado de máquina, entre outros) que analisarão os dados armazenados e permitirão às aplicações (na camada 5) agir de acordo com as configurações do contexto.

Nota-se que nesse tipo de sistema, os modelos de contexto tem um papel fundamental, que vai influenciar, em variado grau, todas as camadas da infra-estrutura. Em vista dessa notável importância, na seção 3.1.2 são tratados tópicos sobre modelagem de contexto em ambientes de computação ubíqua para viabilização de sistemas sensíveis ao contexto.

3.1.2 Modelos de contexto

A modelagem de contexto objetiva a obtenção de descrições formais ou semi-formais das informações de contexto presentes em um sistema sensível ao contexto. As instâncias do modelo de contexto são interpretações de dados brutos obtidos (sentidos) a partir de diversas fontes. Com efeito, os modelos de contexto promovem o compartilhamento e intercâmbio de informações entre as aplicações, viabilizando integração, consulta e raciocínio sobre dados heterogêneos.

Esses requisitos tornam o tratamento de informações contextuais bastante complexo e tema recorrente de pesquisas. O problema de captura do contexto, por si só, já é um impulsionador de investigações em redes de sensores e mineração de fluxos contínuos de dados, entre outros temas relevantes.

Contexto é de difícil modelagem e essa dificuldade, aliada a evidente polissemia desse conceito, desfavorece o estabelecimento de modelos genéricos e universais de contexto. De modo que esses fatores costumam direcionar os modelos ao atendimento de propósitos específicos, como smart-homes [Chen et al. 2003], contexto de entrega de conteúdos [Santana et al. 2007c], contexto de ambientes colaborativos [Wang et al. 2006], contexto de tarefas de usuário [Fischer and Ye 2001], entre os mais recorrentes.

Do ponto de vista de formalismos para modelagem, diversas alternativas vêm sendo propostas [Strang and Linnhoff-Popien 2004]: pares associativos, modelos relacionais, análise multidimensional [Adomavicius et al. 2005], linguagens de marcação, modelos baseados em objetos, modelos baseados em grafos [Mostefaoui et al. 2004], representações lógicas.

Limitações nesses formalismos, principalmente quanto a poder de expressão, têm conduzido ao estabelecimento de *ontologias* como o formalismo mais difundido para modelagem de contexto. Em vista das vantagens oferecidas pelas ontologias, e também em vista da importância desse formalismo nesta dissertação, a seguir serão fornecidos maiores detalhes dos modelos baseados em ontologias.

Modelos de contexto baseados em ontologias

Uma ontologia, no âmbito da Ciência da Computação, é a especificação explícita de uma conceituação compartilhada [Gruber 1995], pertinente a um domínio de discurso; uma conceituação é uma representação abstrata de conhecimento, enquanto uma ontologia é a expressão, dependente de linguagem, de uma conceituação [Guarino 1998]. As ontologias mostram-se úteis para representar e compartilhar o conhecimento acerca de um domínio de discurso, já que possibilitam a formalização de vocabulários de forma explícita, expressiva e com semântica bem-definida, tornando-as adequadas à interpretação semântica por computadores e sujeitas à inferência automática por meio de raciocinadores (*reasoners*).

Essas vantagens são especialmente úteis aos sistemas sensíveis ao contexto, uma vez que viabilizam a interoperabilidade semântica entre fontes de contexto, integração de fontes a informações do domínio e raciocínio automático sobre as instâncias de dados de contexto. As ontologias também são largamente aplicadas em diversos outros domínios que demandam representação de conhecimento, como integração de sistemas, modelagem de software, indexação de recursos, entre outros. Em particular, ontologias têm recebido grande ênfase em pesquisas que envolvem tecnologias de Web Semântica [Lee et al. 2001, Shadbolt et al. 2006], tecnologias dentre as quais destacam-se RDF (*Resource Description Framework*) [Lassila et al. 1999] e OWL (*Web Ontology Language*) [McGuinness et al. 2004].

A variedade de modelos de contexto baseados em ontologia, no âmbito das pesquisas em Computação Ubíqua, é bastante ampla, com modelos que buscam tratar desde *smart-homes* até tarefas de usuários. Em particular, os modelos de contexto que englobam o domínio de trabalho colaborativo em ambientes de computação ubíqua mereceram maior atenção nesta dissertação, devido à direta aplicabilidade a este projeto.

Nas ontologias de contexto em trabalho colaborativo, costuma-se agregar tanto características do usuário quanto do ambiente físico em que o usuário está inserido. Por exemplo, em [Hu and Moore 2007] apresenta-se uma ontologia de contexto aplicável a aprendizado colaborativo assistido por computador que é composta por uma hierarquia de conceitos para modelagem de usuário (identidade, preferências) e características do ambiente físico (localização, dispositivos de acesso) e comportamentais (tarefas realizadas, entre outras); agentes inteligentes raciocinam sobre essa ontologia para recomendar objetos de aprendizagem. Já em [Yang et al. 2006] há a preocupação com

aplicação de modelos de ontologia para representar contexto em ambientes de aprendizado ubíquo, com ênfase no emprego de dispositivos móveis como meio de acesso; destacam-se conceitos para modelar tarefas e colaboradores do usuário.

Uma abordagem semelhante às anteriores é apresentada [Bouzeghoub et al. 2007], preocupando-se com o uso de ontologias de contexto na seleção e adaptação de recursos de aprendizado. Uma contribuição importante deste trabalho é a distinção formal entre contexto (conjunto de conceitos e instâncias da ontologia) e situação (conjunto de contextos sob restrições temporais). Em [Petersen and Cassens 2006] busca-se transferir a teoria das atividades, que trata aspectos sócio-filosóficos e psico-cognitivos da atividade humana, para o problema de modelagem de contexto. Embora os meios teóricos empregados divergem dos outros trabalhos, a ontologia resultante é bastante similar, salvo por alguns conceitos que remontam a aspectos humanos, tais como estados psicológicos e mentais do usuário e contexto social.

Em [Wang et al. 2006] são apresentadas descrições bastante breves de alguns conceitos de uma ontologia de contexto para ambientes colaborativos (de forma geral, modela-se usuários, atividades e conteúdos). Uma perspectiva interessante desse trabalho é a identificação de requisitos para a modelagem de contexto nesses ambientes, perspectiva semelhante à adotada em [Bolchini et al. 2007]. Em particular, destacam-se os requisitos de *memória de contexto* e *similaridade de contexto*, os quais trazem dificuldades adicionais em modelos baseados em ontologias.

O requisito de memória de contexto especifica que algumas instâncias do modelo de contexto são válidas apenas durante um período específico de tempo. Quando considera-se ontologias em OWL, especificamente, é importante ressaltar a existência de restrições técnicas para cumprir esse requisito, principalmente por não haver recurso eficiente para expirar fatos de uma ontologia em OWL. Para esse fim, têm sido propostos meios de operar com grafos nomeados para tratar esse requisito [Carroll et al. 2005, Bouquet et al. 2005, Stoermer et al. 2006].

Outro requisito importante é a similaridade entre contexto, mais especificamente, similaridade entre instâncias do modelo de contexto. Diversas pesquisas tem sido empreendidas na direção de estabelecer métricas e estratégias de similaridade para ontologias. Em [Maedche and Staab 2002] apresenta-se um arcabouço teórico para calcular a similaridade entre ontologias considerando métricas para similaridade sintática e semântica. Uma abordagem análoga pode ser encontrada em [Ehrig et al. 2005], porém considerando também similaridade pragmática e algumas aplicações. Objetivando a

integração de métricas de similaridade em linguagens de consulta para ontologias, em [Kiefer et al. 2008] estende-se a linguagem normativa de consulta SPARQL [Pérez et al. 2006] para integrar estratégias de similaridade nas consultas em bases de dados RDF.

Expostos os principais aspectos das pesquisas de contexto em Computação Ubíqua, a próxima seção trata a noção de contexto construída sob a Recuperação Interativa de Informação.

3.2 Contexto em Recuperação Interativa de Informação

Tradicionalmente, aplicações para Recuperação de Informação (RI) têm adotado uma perspectiva dita "centrada no sistema" em que a qualidade do sistema é mensurada segundo sua efetividade algorítmica, desconsiderando a influência do usuário no processo de busca. Recentemente, tem sido freqüente a defesa de abordagens mais "centradas no usuário" para o desenvolvimento de sistemas de RI, que considerem as influências que a subjetividade do usuário exerce sobre suas necessidades informacionais [Ingwersen and Järvelin 2005] [Järvelin and Ingwersen 2004] [Järvelin 2007] [Ingwersen 1992b].

Essas abordagens, comumente agregadas sob a Recuperação *Interativa* de Informação, visam tornar os sistemas de RI mais adaptativos e personalizáveis. As demandas por abordagens que considerem aspectos humanos em recuperação de informação têm sido disseminadas devido principalmente a três mudanças de perspectiva ocorridas na literatura de RI, que buscaram novas interpretações para caracterizar necessidades informacionais:

- a *perspectiva da relevância*, caracterizada por uma gradual aceitação de que uma requisição de informação (consulta) não corresponde diretamente a uma necessidade informacional. Consultas são vistas como expressões potencialmente incompletas e parciais que podem desfavorecer o atendimento de fatores psico-cognitivos e culturais (contexto) da necessidade informacional. Com isso, defende-se que a relevância de um objeto informacional deve ser julgada em relação à necessidade informacional, e não em relação à consulta.

- a *perspectiva cognitiva*, que compreende que a necessidade informacional é reflexo de um estado anômalo de conhecimento (*Anomalous State of Knowledge* — ASK).

Um estado de conhecimento consiste em uma lacuna no conhecimento do usuário que o leva a interagir com algum sistema de RI; um estado de conhecimento torna-se anômalo quando o usuário, devido à gravidade da lacuna, não consegue satisfatoriamente interrogar o sistema (não consegue antever que termos poderiam ser usados na consulta). Em geral, isso ocorre pois o usuário não consegue contextualizar sua necessidade frente ao domínio de conhecimento investigado.

- a *perspectiva interativa*, que amplia o processo de recuperação de informação como uma sessão interativa. A interação com um sistema de RI traz dinamicidade à forma como o usuário julga a relevância dos documentos, pois o contato em tempo real com os objetos informacionais traz novos pontos de vista à forma como o usuário entende o domínio. Sendo assim, o sistema não deve considerar apenas o casamento consulta-documentos, mas também considerar as interações do usuário com o sistema, com documentos, e também o comportamento das reformulações de consulta.

Nota-se por essas perspectivas que as necessidades informacionais não podem ser consideradas como abstrações estáticas e auto-contidas numa consulta, mas que é preciso considerar também fatores externos, que permitam caracterizar o *contexto*, ou situação, do usuário no momento da busca. Ao negligenciar esses fatores, o desempenho do sistema é em grande parte relegado à capacidade do usuário em formular requisições de informação precisas. Desse modo, dependendo do grau de conhecimento prévio do domínio de discurso e dos mecanismos do sistema, bem como da complexidade da necessidade informacional, a interação com o sistema de RI pode constituir uma sobrecarga cognitiva para o usuário.

Esta constatação ilustra o fato de que usuários de sistemas interativos para acesso à informação podem manifestar habilidades distintas para especificar necessidades informacionais dependendo da complexidade e da estruturação da tarefa em que estão envolvidos [Järvelin and Ingwersen 2004] [Freund et al. 2005] [Byström and Hansen 2005] bem como no conhecimento do domínio que estão investigando. De fato, a especificação de uma necessidade informacional é fortemente influenciada por fatores *contextuais*, que direcionarão como esses usuários avaliarão a relevância dos documentos recuperados. Nota-se com isso que a tradução de uma necessidade informacional em uma consulta, quando fracamente contextualizada, pode conduzir à representação incompleta das intenções originais do usuário [Li and Belkin 2008].

De forma geral, quando surge uma necessidade informacional, o usuário está en-

volvido em alguma tarefa mais ampla — denominada tarefa de trabalho — que demanda informações complementares para ser cumprida. Uma tarefa de trabalho é uma atividade profissional ou cotidiana a ser executada pelo usuário. Se a tarefa de trabalho não pode ser imediatamente cumprida devido a falta de informação, isto pode caracterizar um estado de incerteza que conduz o usuário a interagir com os canais de informação disponíveis, entre os quais incluem-se os sistemas de RI [Ingwersen and Järvelin 2005]. Este comportamento se deve ao fato de que certas tarefas de trabalho — como as executadas em ambientes ricos em conhecimento — inerentemente envolvem processamento de informação em grau considerável.

As tarefas de trabalho e a própria interação do usuário com o sistema de RI incluem parâmetros contextuais que influem diretamente nas intenções de uma necessidade informacional [Byström and Hansen 2005]. A figura 3.2 ilustra as principais dimensões de contexto presentes nesse cenário.

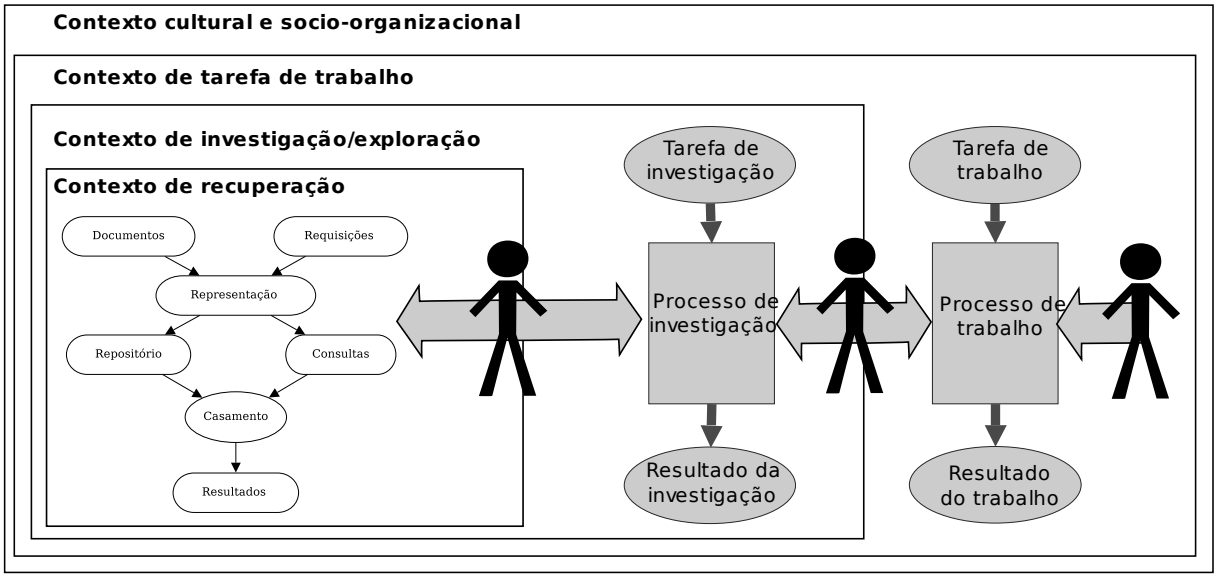

Figura 3.2: Níveis de contexto em RI interativa. Adaptado de [Järvelin and Ingwersen 2004] e [Ingwersen and Järvelin 2005]

Como pode ser notado pela figura 3.2, o processo de busca por informação está embutido em níveis incrementalmente sofisticados de contexto [Järvelin and Ingwersen 2004]. Na forma mais ampla, o usuário está inserido em um contexto cultural e sócio-organizacional que vai influenciar como são compreendidos os problemas a serem tratados. Esse contexto inclui o contexto da tarefa de trabalho, que indica em que o usuário está envolvido quando a necessidade informacional surge. A obtenção de resultados na tarefa de trabalho é o motivo principal da busca por informação, o que levará o usuário a uma tarefa de investigação, na qual ocorrerá a conversão da neces-

sidade informacional numa consulta. A partir desse momento, o usuário pode iniciar uma tarefa de recuperação, que consistirá na interação efetiva com o sistema de RI.

Nota-se com isso que o contexto influencia fortemente a forma como um usuário compreende sua necessidade informacional. A abordagem centrada no sistema em grande parte ignora essa riqueza de contexto, delegando ao usuário toda a sobrecarga cognitiva da formulação de consulta. Como efeito colateral, usuários em diferentes contextos, mas que formulem a mesma consulta, serão apresentados a resultados idênticos. De forma a melhorar esse cenário generalista, um sistema que explore representações de atributos das tarefas de trabalho tem potencial para melhor satisfazer as necessidades informacionais. Isto porque o sistema pode contar com subsídios adicionais para interpretar o que permeia a consulta, ao invés de concentrar-se exclusivamente no tema da necessidade informacional.

Abordagens para explorar contexto em RI interativa aglutinam-se principalmente em esforços para caracterizar a tarefa de trabalho do usuário enquanto interage com o sistema de RI e investigação de formas pelas quais estas caracterizações podem influenciar o processo de busca. O sistema apresentado em [Hernandez et al. 2007] modela tarefas executadas por astrônomos de forma a viabilizar a construção de uma interface contextual de navegação de resultados. Em [Freund and Toms 2005] é apresentada uma abrangente elicitação de tarefas de trabalho executadas por engenheiros de software objetivando a especificação de requisitos para um sistema de busca contextual. Embora os requisitos apontados revelem-se bastante pertinentes, a proposta mostra-se muito teórica e não esclarece como as tarefas de trabalho podem ser exploradas na prática para obter busca contextual. Tarefas de trabalho de engenheiros aeronáuticos são consideradas em [Redon et al. 2007], por meio de modelo de contexto orientado a objetos. Os metadados representados por esse modelo são posteriormente usados para recomendar quais documentos são relacionados ao contexto atual; um ponto fraco emerge do fato de que o usuário deve prover explicitamente esses metadados.

Esta seção expôs a noção de contexto trabalhada nas pesquisas de Recuperação Interativa de Informação. Na próxima seção, será tratado como contexto é abordado em pesquisas de personalização. Adicionalmente, será introduzida uma técnica importante para trabalhar com dados contextuais: o raciocínio baseado em casos.

3.3 Contexto em personalização

De forma geral, personalização abrange um conjunto de técnicas comumente empregadas para adaptar o acesso à informação de acordo com interesses e preferências dos usuários ou grupos de usuários. As técnicas de personalização partem da modelagem de preferências e interesses e aplicam os modelos em sistemas adaptativos capazes de ajustar suas aparências ou comportamentos [Brusilovsky et al. 2007]. As pesquisas em personalização são numerosas e fragmentadas, envolvendo diferentes comunidades de pesquisa, como hipermídia, aprendizado de máquina, recuperação de informação, inteligência artificial, entre outras. Também bastante variadas são as aplicações de personalização, entre as quais destacam-se como principais grupos:

- **Busca personalizada**: busca personalizada ocorre quando o sistema de RI acumula um histórico de consultas e documentos com os quais o usuário interagiu e o aplica para refinar recuperações futuras. Isso envolve modelagem, aprendizado e coordenação de modelos de usuário, abrangendo preferências de curto ou longo prazos, isto é, preferências do usuário ao longo do tempo. Dessa forma, dois usuários com a mesma consulta deparam-se com diferentes resultados pois os contextos apreendidos pelo sistema diferem para os dois sujeitos. Outros avanços interpretam o contexto como as interações sociais em comunidades de usuários, sob abordagens de filtragem colaborativa e RI para redes sociais [Sieg et al. 2007, Gauch et al. 2003].

- **Hipermídia adaptativa**: consiste na personalização de sistemas hipermídia por meio de transformações no espaço informacional, que é visto como um grafo. As transformações abrangem *navegação adaptativa*, em que os documentos são reorganizados para prover uma navegação personalizada no espaço informacional; *apresentação adaptativa*, em que o conteúdo dos documentos é adaptado, por exemplo para atender a restrições do dispositivo de acesso; *seleção adaptativa*, em que são recomendados pontos de partida para o usuário iniciar a exploração do espaço informacional [Brusilovsky 2003].

- **Filtragem e recomendação**: sistemas de filtragem agem como intermediários entre um fluxo de documentos e grupos de usuários, redirecionando documentos seletivamente de acordo com o grau de casamento entre os documentos e as preferências explícitas dos usuários [Hanani et al. 2001]; sistemas de recomendação mais comumente trabalham com informações implícitas e buscam sugerir docu-

mentos enquanto o usuário interage com o sistema [Adomavicius and Tuzhilin 2005].

Uma característica comum a estes sistemas personalizados é a coleta de dados acerca de seus usuários por meio de monitoramento implícito da interação com os objetos informacionais ou mesmo requisitando dados explicitamente para o usuário. Os dados assim obtidos são atributos contextuais formalizados em modelos de usuários (ou perfis), que são usados pelo mecanismo de adaptação do sistema para ajustar seu comportamento ou aparência. Os perfis são vistos como modelos de contexto, que buscam caracterizar a situação do usuário a partir das interações com o sistema.

Em [Sieg et al. 2007] são utilizados atributos contextuais de interação (monitoramento de cliques) para personalizar buscas. Os perfis são representados como subconjuntos de uma ontologia de domínio, cujos conceitos recebem pesos de acordo com preferências de curto e longo prazo do usuário. O efeito do sistema é a reordenação do *ranking* de resultados de acordo com o perfil do usuário, usando um mecanismo de propagação de ativação na ontologia. Já em [Mylonas et al. 2008] os perfis de usuário são baseados em lógica difusa, englobando consultas prévias, documentos navegados, etc. Os modelos são usados para identificar, numa ontologia de domínio, conjuntos difusos de interesses do usuário que englobam regiões da ontologia. Os interesses do usuário assim obtidos são utilizados em uma adaptação do modelo de espaço de vetores que irá realizar o *matching* considerando também as informações dos perfis.

Em [Barbosa et al. 2007] é descrito um sistema para filtragem de objetos de aprendizado baseando-se em perfis de contexto. Os perfis são construídos utilizando-se padrões de metadados educacionais e a filtragem ocorre de acordo com interesses e localização física dos usuários. Já em [Rigo and Oliveira 2007] é descrito um sistema de hipermídia adaptativa apoiado em mineração de uso e ontologias de domínio para adaptar a navegação em websites, especificamente provendo menus adaptativos e dicas de navegação. Em [Silva and Favela 2006] é descrito um sistema para busca personalizada de informações médicas. O perfil de usuário é basicamente o prontuário pessoal do paciente, que contém informações demográficas e histórico de saúde. Os dados presentes no prontuário são usados para filtrar documentos ou alterar o *ranking* de resultados.

A seguir é enfatizada uma técnica amplamente usada em personalização e raciocínio sobre informações de contexto: o raciocínio baseado em casos.

Raciocínio Baseado em Casos

O Raciocínio Baseado em Casos (RBC, *Case Based Reasoning*) é uma metodologia para aprendizado de máquina que baseia-se nos conceito de analogia e de similaridade, em particular na noção de que situações reconhecidas como similares em certos atributos podem ser análogas. Deste modo, RBC configura-se como uma forma simples de raciocínio por analogia que pode ser aplicada em métodos de resolução de problemas [Kolodner 1993]. A idéia principal do RBC é a solução de novos problemas com base em experiências, que são representadas por problemas do mesmo tipo que foram previamente resolvidos. Tais problemas são denominados casos e um novo caso é resolvido adaptando-se a solução de um caso similar [Aamodt and Plaza 1994].

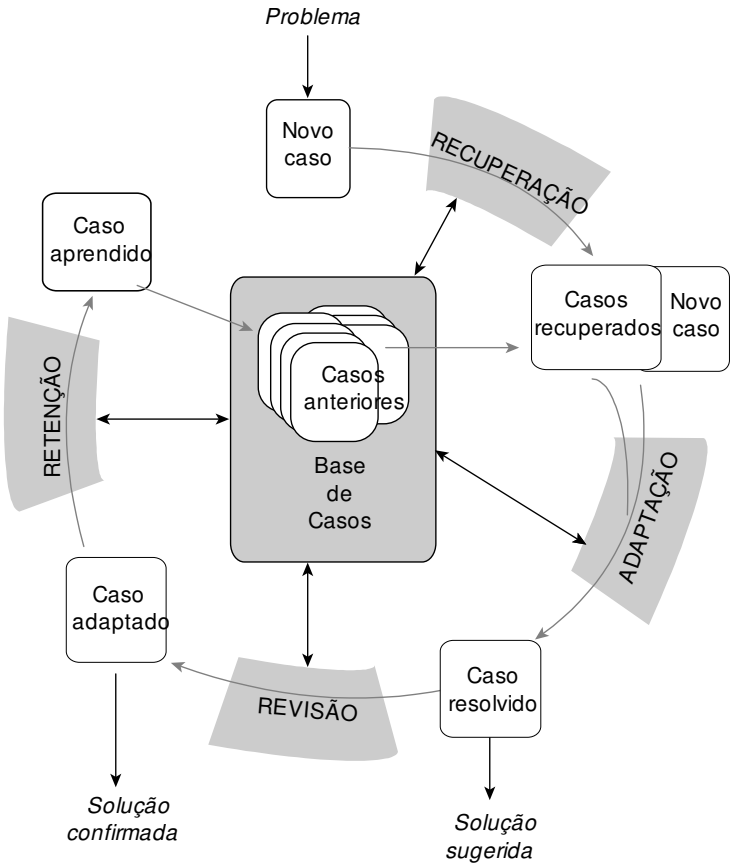

Figura 3.3: Ciclo típico de RBC. Adaptado de [Pal and Shiu 2004]

Um sistema que empregue RBC requer técnicas eficientes para concretizar as subtarefas que compõem a metodologia, a exemplo da organização e manutenção da base de casos, da recuperação de casos similares a partir da base de casos e da adaptação dos casos recuperados ao problema atual. O mecanismo básico de inferência do RBC é fundamentado no princípio de aprendizado de máquina baseado em instâncias, bem como em métricas de similaridade entre as instâncias [Mitchel 1997]. A figura 3.3 provê

uma visão geral de um ciclo típico de RBC.

Um caso é composto por atributos do problema e por uma solução do problema. Os atributos são usados em métricas de similaridade e, pela comparação de vários casos, permitem que se obtenham soluções por analogia. Como pode ser notado pela figura 3.3, um ciclo de RBC é composto por quatro passos:

1. **Recuperação**: formulado o problema em um novo caso, recuperar casos passados que sejam similares ao novo caso;
2. **Adaptação**: a partir dos requisitos do novo caso, adaptar os casos recuperados para o novo problema, obtendo uma solução sugerida;
3. **Revisão**: a solução sugerida é revisada para averiguar se é satisfatória; caso seja satisfatória, a solução é confirmada; do contrário, a solução deve ser rejeitada;
4. **Retenção**: o caso adaptado é persistido na base de casos para ser utilizado em futuros raciocínios.

Os primeiros sistemas de RBC foram usados numa variedade de tarefas para resolução de problemas e classificação de dados, destacando-se em relação aos métodos tradicionais até então empregados, por basearem-se em experiências concretas ao invés de conhecimento na forma de regras e modelos rígidos de domínio. Atualmente, RBC tem sido usado em personalização via sistemas de recomendação e raciocínio sobre informações de contexto. Em sistemas de recomendação, os objetos informacionais são representados como casos e recomendações são geradas por meio da recuperação dos casos que mais se assemelham a uma consulta ou perfil do usuário [Brusilovsky et al. 2007]. Já em raciocínio sobre informações de contexto, os dados de contexto são usados para compor os casos delimitados por restrições temporais, formando uma memória de contexto. A aplicação da metodologia de raciocínio habilita os sistemas sensíveis ao contexto a aprender as reações mais adequadas de acordo com características situacionais do ambiente [Petersen 2006].

Em [Redon et al. 2007] é apresentada um aplicação que emprega RBC para recomendar documentos a usuários de sistemas de projetos em engenharia. Os casos foram modelados como objetos e as métricas de similaridade operam sobre os atributos dos objetos. Preocupado com a recomendação de músicas em ambientes de Computação Ubíqua, em [Lee and Lee 2008] os casos são construídos a partir de dados provenientes de sensores do ambiente, e englobam atributos que permitem recomendar músicas

com base em demografia, preferências e também similaridade dos ambientes físicos em que as músicas foram ouvidas. Já em [Aktas et al. 2004] é apresentado um sistema que emprega RBC para recomendar metadados de simulação em ambientes de *grid*. Uma contribuição importante deste trabalho é a representação dos casos, que são formados por coleções de instâncias de ontologias de domínio. Ontologias também são empregadas para representar os casos em [Bai et al. 2008], porém voltando-se ao raciocínio sobre dados de contexto (tempo, localização, etc.) em ambientes de Computação Ubíqua.

Diversos requisitos para uma aplicação de RBC no problema de raciocínio sobre informações de contexto são apresentados em [Zimmermann 2003], com ênfase em uma aplicação em guias inteligentes de museu. São apresentadas alternativas para representação de atributos de contexto na forma de casos (desde representações relacionais, orientadas a objetos e baseados em grafos) e adaptações da metodologia para situá-la nesta classe de problemas. Uma abordagem análoga é apresentada em [Petersen 2006], voltada a uma aplicação de guia turístico para dispositivos móveis. São apresentadas alternativas para a representação dos casos considerando atributos de contexto e aponta-se a necessidade de empregar restrições temporais para contextualizar os atributos.

Conforme apresentado na seção 3.1.2, modelos de contexto em sua maioria atendem a propósitos específicos e demandam requisitos como memória de contexto (manutenção de um histórico de estados temporais dos atributos de contexto) e similaridade entre itens armazenados na memória de contexto. RBC revela-se como uma metodologia adequada para tratar ambos os requisitos, uma vez que permite alimentar uma base de casos ao longo do tempo e executar consultas de similaridade nos casos. Em específico, quando os casos são modelados usando ontologias, RBC apresenta vantagens sobre o raciocínio tradicional baseado em regras, pois não exige a definição manual e *apriori* de um conjunto expressivo de regras [Bolchini et al. 2007].

A metodologia RBC pode ser aplicada em um problema de forma *ad-hoc* ou a partir da especialização de infra-estruturas que englobem as diversas tarefas necessárias ao raciocínio. Exemplos dessas infra-estruturas são os *frameworks* FreeCBR[1] e jColibri[2]. Em particular, o *framework* jColibri apresenta ferramentas para projetar RBC e APIs (*Application Programming Interfaces*) para integrar a metodologia às aplicações. Adicionalmente, jColibri dispõe de funcionalidades para uso de ontologias na repre-

[1] `http://freecbr.sourceforge.net/`
[2] `http://gaia.fdi.ucm.es/projects/jcolibri/`

sentação de casos e conta com uma biblioteca de métricas de similaridade para ontologias [Recio-Garcıa et al. 2006].

Nesta seção foram expostos os principais aspectos para se lidar com contexto em personalização e foi apresentada uma técnica, muito empregada em sistemas personalizados, que mostra-se útil para raciocinar com dados contextuais. A próxima seção encerra este capítulo.

3.4 Considerações finais

Este capítulo expôs a importância do contexto das necessidades informacionais para determinar a relevância dos documentos recuperados em um sistema de RI. Foram apresentadas as principais tendências de pesquisa para se trabalhar com contexto, com ênfase em três delas: Computação Ubíqua, Personalização e Recuperação Interativa de Informação. Nesse ínterim, tratou-se de aspectos teóricos, bem como sistemas sensíveis ao contexto, abordagens para modelagem de contexto e raciocínio baseado em casos aplicado a dados contextuais.

O próximo capítulo tratará de uma importante extensão dos modelos de RI, que visa minimizar dificuldades de contextualização no processo de formulação de requisições de informação: a modificação de consulta. Dentre as técnicas de expansão de consulta, será enfatizada a retroalimentação de relevância, especialmente útil para a aplicação de dados contextuais.

4 *Modificação de consulta e retroalimentação de relevância*

Em grande parte das coleções de documentos, um mesmo conceito pode ser expresso por diferentes formas léxicas, dificultando ao usuário formular requisições de informação com alto grau de fidelidade em relação ao vocabulário do *corpus*. Esse fenômeno, denominado divergência (ou impedância) de vocabulário, é bastante recorrente e degrada a revocação do sistema. Tipicamente, ocorre quando o usuário apresenta um estado anômalo de conhecimento (*Anomolous State of Knowledge* ou ASK), ou seja, apresenta dificuldade de antever os tópicos dos conteúdos e baixa familiaridade com o domínio de discurso indexado pelo sistema.

As consultas nos sistemas de RI atuais, principalmente os baseados na Web, são não-estruturadas e curtas. Estes fatores, embora ampliem a usabilidade do sistema de RI, tendem a promover consultas ambíguas e fracamente contextualizadas. Dentre os tipos de consulta emitidas nestes sistemas, destacam-se três categorias principais [Manning et al. 2008]:

- **consultas informacionais:** expressam temas gerais e abrangentes, como "tipos de diabetes" ou "turismo em Recife". Tipicamente, a coleção não conterá um único documento que por si só satisfaça a necessidade informacional. Consequentemente, um comportamento típico de usuários com consultas informacionais é a leitura de vários documentos recuperados, até que se considere satisfeita a necessidade informacional ou mude-se o foco da tarefa de busca. Por essa razão essas consultas são também conhecidas como consultas de pesquisa (*research queries*).

- **consultas navegacionais**: ocorrem quando o usuário não recorda exatamente o percurso necessário para chegar a um documento, como por exemplo "roteiro mapas conceituais unesp" ou "secretaria fazenda sp". Nesse caso, o usuário assume *a priori* a existência do documento e interroga o sistema com a intenção

de obter a localização desse recurso. Muito comum em ambientes nos quais a organização dos documentos é flexível (e.g. a Web, em que muitas consultas navegacionais ocorrem quando o usuário não recorda o endereço completo de um *website*). Neste tipo de consulta, o usuário espera encontrar apenas um documento relevante, próximo ao topo da lista de resultados.

- **consultas transacionais:** são consultas que antecedem a efetivação de alguma transação online pelo usuário. Mais comum em ambientes que provêm serviços, como a Web, tais consultas antecedem ações do usuário como comprar produtos, descarregar arquivos, fazer reservas em hotéis, etc. Neste caso, o usuário espera que o sistema ofereça indicadores de serviços que permitam realizar a ação desejada.

Para aliviar os problemas de formulação de consultas, uma possível solução é modificar a consulta original, reduzindo-a ou, mais comumente, expandindo-a, com termos relacionados, em processos automáticos ou interativos. O presente trabalho dedicará maior atenção aos métodos que modificam a consulta pelo acréscimo de novos termos, isto é, aos métodos de expansão de consulta.

Com os métodos de expansão, o usuário é guiado a formular consultas que obtenham resultados de maior grau de utilidade às suas necessidades informacionais. Isso é conseguido por um processo que reformula a consulta original, acrescentando novos termos, cujos significados possuem alguma relação semântica — latente ou explícita — com os termos originais. Tal processo pode ser assistido pelo usuário, que escolhe termos a partir de alternativas de expansão computadas pelo sistema; ou automático, durante o qual o usuário não é conscientemente envolvido no processo.

Aquém da estratégia adotada pelo processo de expansão, o objetivo final é a obtenção de uma consulta informacionalmente contextualizada, com o fim de minimizar a divergência de vocabulário entre requisição de informação e coleção de documentos. Tais subsídios contextuais podem ser providos com assistência do usuário, ou por análises de co-ocorrência entre os termos da coleção ou também por fontes de vocabulário controlado e modelos de representação de conhecimento.

Metodologicamente, o processo de expansão pode agir por análises globais, ou seja, independentemente dos resultados recuperados, operando a partir de fontes de vocabulário, que podem ser estáticas, ou dinamicamente derivadas do *corpus*; ou a partir de análises locais, ajustando a consulta a partir de evidências extraídas diretamente

dos documentos recuperados pela consulta original [Muresan 2006]. Dentre os principais métodos de expansão de consulta, destacam-se [Xu and Croft 1996, Billerbeck and Zobel 2004, Bhogal et al. 2007]:

Expansão baseada em vocabulário controlado: nesse método global, os termos de expansão provêm de fontes de vocabulário controlado (e.g. taxonomias, *thesauri*) ou modelos de representação de conhecimento (e.g. ontologias, redes semânticas, redes léxicas) criados intelectualmente por especialistas. A inclusão de novos termos pode ser realizada manualmente pelo usuário, a partir de sugestões do sistema, ou automaticamente, por algoritmos de análise probabilística.

Expansão baseada em construção de vocabulário: método global bastante similar à expansão por vocabulário controlado. Porém, os termos de expansão provêm de estruturas geradas automaticamente por algoritmos de processamento de texto. A fonte mais comum são *thesauri* construídos automaticamente pela análise de co-ocorrência de termos nos documentos da coleção.

Correção ortográfica: método local de modificação de consulta que busca identificar erros ortográficos acidentalmente cometidos pelo usuário durante a formulação da consulta original. De forma geral, são apresentados ao usuário prováveis erros e sugeridas consultas reformuladas que os eliminem.

Retroalimentação de relevância: nesse método local, os termos de expansão são seletivamente extraídos dos resultados da consulta original. A decisão de quais documentos participarão do processo pode determinar-se por um limiar de corte sobre o *ranking* de resultados (*top-k threshold*) ou, alternativamente, por documentos apontados explicitamente pelo usuário, em tempo de interação. Em meio a todo o vocabulário dos documentos assim selecionados, computam-se apenas alguns termos como elegíveis para expansão.

Doravante serão dadas maiores vistas aos métodos de expansão baseados em retroalimentação de relevância.

4.1 Retroalimentação de relevância

Retroalimentação de relevância (*relevance feedback*) é um processo de expansão de consulta, baseado em análise de conteúdo informacional, que guia-se por evidências

de quais documentos são relevantes para a consulta, de forma a subsidiar a eleição de termos de expansão [Ruthven and Lalmas 2003]. Apesar de as pesquisas em retroalimentação de relevância acumularem mais de três décadas de desenvolvimentos, suas possibilidades de incorporar ganhos qualitativos nos sistemas de RI ainda são tema de intensa investigação científica. Em respeito às evidências consideradas no processo, as estratégias de retroalimentação podem ser classificadas em três grupos: retroalimentação explícita de relevância (*explicit relevance feedback*); retroalimentação cega de relevância — ou pseudo-retroalimentação de relevância (*blind/pseudo relevance feddback*); e retroalimentação implícita de relevância (*implicit relevance feedback*).

A idéia central da retroalimentação explícita de relevância é, iterativamente, coletar julgamentos — negativos ou positivos — de relevância, do usuário, expandindo cada iteração de consulta com discriminadores dos documentos escolhidos, até que a necessidade informacional seja satisfeita. A figura 4.1 ilustra essa situação.

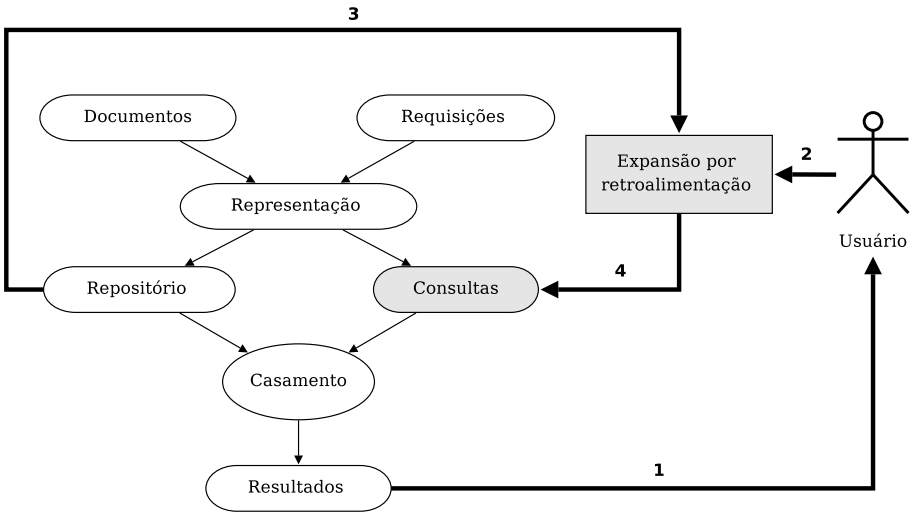

Figura 4.1: Retroalimentação explícita de relevância.

Na figura 1, assume-se que o processo iniciou-se com a emissão ao sistema de uma requisição de informação inicial (evento não exposto na figura), resultando numa listagem inicial de resultados. A partir desse momento, ocorrem os seguintes passos:

1. O usuário analisa a listagem de resultados de forma a julgar quais ítens são relevantes à sua requisição;

2. Os julgamentos são submetidos ao sistema;

3. Os documentos julgados têm seu conteúdo informacional analisado por um algoritmo que elege os melhores termos discriminadores desse conjunto, adicionando-

os à consulta original e recalculando os pesos;

4. A consulta reformulada é submetida ao sistema e os resultados são reapresentados.

Esse processo pode-se repetir ao longo de várias iterações até que o usuário considere satisfeita sua necessidade informacional. Tal abordagem apoiada em evidências explícitas é capaz de obter ganhos expressivos [Salton and Buckley 1990], desde que haja significativa participação do usuário no processo. Porém, o provento explícito de julgamentos de relevância constitui, em situações reais, uma sobrecarga cognitiva sensivelmente alta durante as tarefas de busca, desincentivando os usuários a prover os julgamentos; além disso, após várias iterações, há o risco dos resultados afastarem-se do conteúdo da requisição inicial, levando o usuário a desconcentrar-se da sua intencionalidade de busca [Kelly and Teevan 2003]. Apesar destas desvantagens, a retroalimentação explícita ainda é usada em sistemas de recuperação de imagens, nos quais a indexação dos recursos é mais complexa [Traina et al. 2006].

Uma possível alternativa a esse problema é eliminar o usuário do processo, confiando, ao sistema, os julgamentos: trata-se do princípio básico do método de retroalimentação cega de relevância. Nessa estratégia, adota-se a premissa de que os documentos recuperados a partir da consulta original constituem evidências suficientes — embora irreais — para a derivação de discriminadores que servirão à expansão da consulta. A figura 4.2 ilustra essa situação.

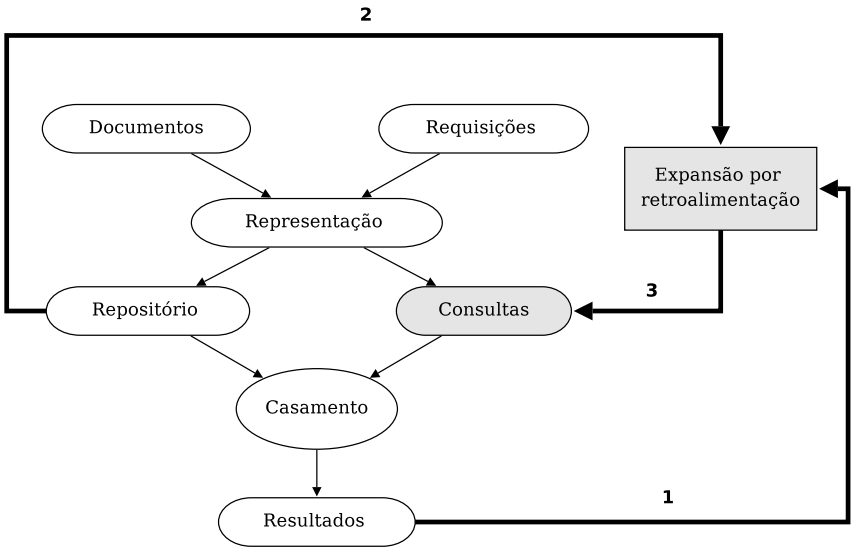

Figura 4.2: Retroalimentação cega de relevância.

A partir da listagem inicial de resultados (1), que não é apresentada ao usuário, o

sistema seleciona um subconjunto de documentos (2), de cardinalidade limitada por um limiar de corte sobre o *ranking* (*top-k threshold*, com $k = 20$ em geral). A partir desse espaço de evidências, são derivados os discriminadores e seus respectivos pesos. Por fim, a consulta expandida é emitida ao sistema (3) e os resultados apresentados ao usuário. Vale ressaltar que o processo, nessas condições, constitui-se por apenas uma iteração.

Com isso, o usuário não participa ativamente do processo de expansão e pode nem mesmo ter consciência de que a consulta foi expandida. Embora seja simples e apresente efeitos positivos sobre o desempenho da recuperação, o sucesso de tal estratégia é condicionada à qualidade da consulta original. Mais precisamente, se a consulta inicial for mal formulada, os primeiros resultados da listagem terão baixa precisão [Baeza-Yates and Ribeiro-Neto 1999]. Em conseqüência, discriminadores pobres serão derivados desses resultados, agravando ainda mais o efeito negativo da consulta original, fenômeno denominado *query drift* [Ruthven and Lalmas 2003]. Em suma, a retroalimentação de pseudo-relevância só é efetiva para consultas com bom grau de fidelidade às necessidades informacionais.

Uma alternativa para aliviar a carga cognitiva do provento de evidências explícitas e também evitar o desempenho variável do emprego de evidências de pseudo-relevância, é a adoção de evidências implícitas durante o processo de expansão. Evidências implícitas são subsídios de natureza contextual que denotam a causa da necessidade informacional e como a informação será usada [Ingwersen and Järvelin 2005, Freund and Toms 2005], obtidas através da análise dos padrões de consumo de informação e interação com o sistema. A figura 4.3 ilustra o processo básico de retroalimentação de relevância implícita.

O processo exposto na figura 4.3 inicia-se quando o sistema apresenta um conjunto inicial de resultados, em resposta a uma requisição do usuário. Enquanto o usuário interage com os resultados, seu comportamento é não-intrusivamente monitorado, de forma a permitir ao sistema derivar evidências que indiretamente denotem quais documentos são relevantes para sua necessidade informacional (2). A partir das evidências coletadas (3), o sistema infere e seleciona (4) quais documentos podem ser julgados como relevantes e extrai seus discriminadores. Por fim, a consulta expandida é emitida ao sistema (5) e os novos resultados são apresentados.

Tal processo reflete a coleta das evidências de relevância provenientes da interação imediata do usuário com os resultados, modalidade clássica da retroalimentação im-

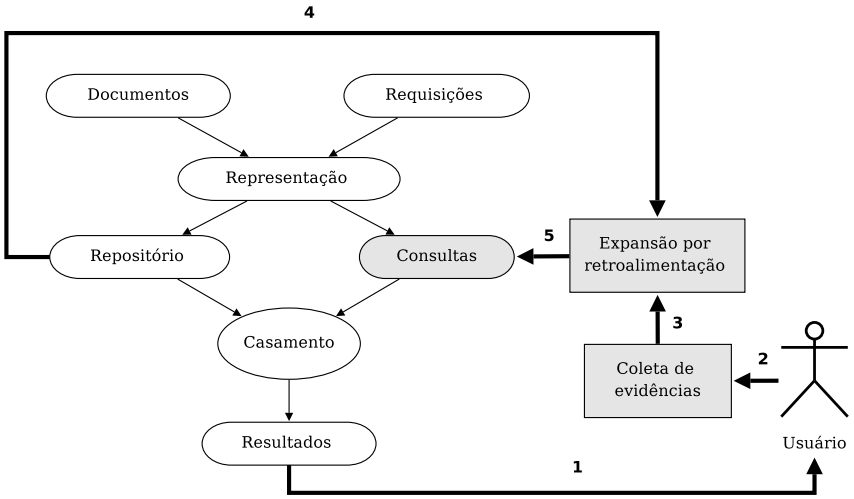

Figura 4.3: Retroalimentação implícita de relevância.

plícita de relevância [Shen et al. 2005b, White and Kelly 2006, White et al. 2004]. Nessa variante, comportamentos contextualizadores sobre os ítens de resultados, comumente denominados *surrogates* de julgamentos de relevância, são monitorados para subsidiar iterações de consulta dentro do intervalo delimitado por uma sessão de busca. Podem-se citar como exemplos desses comportamentos: padrão de cliques (*click-through behavior*), tempo de leitura, *bookmarking*, seleção, impressão, movimento de olhos, entre outros menos explorados [Jung et al. 2007b, Kelly and Teevan 2003, White et al. 2006].

Contudo, devido à natureza contextual das evidências implícitas, outras fontes, embora não tão imediatas, são exploradas na literatura, como informações ambientais (e.g. tempo, espaço, meio de acesso), evidências situacionais (e.g. contexto da tarefa de trabalho que originou da tarefa de busca), modelos ponderados de domínio ou de tópicos (evidenciando preferências) — tratadas no capítulo 3. Nota-se, com isso que, apesar de driblar as desvantagens das retroalimentações explícita e cega de relevância, a retroalimentação implícita de relevância revela-se mais complexa à medida que adota evidências mais difíceis de coletar e processar. Contudo, essa modalidade de retroalimentação é vista como a mais adequada para refletir a natureza dinâmica e multidimensional da relevância [Borlund 2003a] em situações reais.

Na literatura, há diversas abordagens para retroalimentação implícita de relevância, explorando diferentes tipos de evidências. Mineração de cliques a partir da lista de resultados tem sido usada como fonte de evidências em conjunto com análise de histórico de consultas [Shen et al. 2005a]. Uma abordagem mais abrangente amplia a monitoria de cliques além da lista de resultados, englobando também todos os documentos navegados dentro de uma sessão de busca delimitada por um intervalo de

tempo [Jung et al. 2007a]. Afora os documentos diretamente relacionados a uma sessão de busca online, em [Teevan et al. 2005] os dados indexados na estação de trabalho do usuário são explorados para elicitar seus temas prediletos. Independentemente dos tipos de evidências empregadas, todas essas abordagens tentam aproximar o contexto do usuário baseando-se tão-somente na interação com os tópicos dos documentos, desprezando qualquer conhecimento da tarefa que o usuário está executando.

Aquém do tipo de evidência adotada, um fator importante do processo de retroalimentação de relevância é o algoritmo para seleção de discriminadores de documentos e recálculo de pesos dos termos da consulta. Vários algoritmos para esse fim podem ser encontrados na literatura, como Rocchio, Idec hi, RSV, EMIM [Vinay et al. 2005, Wong et al. 2008]. O algoritmo de Rocchio [Rocchio 1971] é aplicável ao modelo de espaço de vetores e define a expansão como um problema de obtenção de uma consulta ótima. A otimalidade da consulta consiste em maximizar a diferença entre o vetor médio dos documentos relevantes e o vetor médio dos documentos não-relevantes.

Devido a um estado anômalo de conhecimento, o usuário pode não conseguir formular uma consulta ótima. O algoritmo de Rocchio busca mover o vetor de consulta do usuário para uma posição mais próxima da região dos documentos relevantes e, consequentemente, mais distante da região dos documentos não-relevantes. Isto é conseguido por meio da adição e ajuste de termos de forma a melhor discriminar entre documentos relevantes e não-relevantes. O algoritmo de Rocchio é formulado da seguinte maneira:

$$q_e = \alpha q_0 + \beta \frac{1}{|R|} \sum_{i=1}^{|R|} R_i - \gamma \frac{1}{|S|} \sum_{i=1}^{|S|} S_i$$

Nesta formulação, q_0 é o vetor da consulta original, q_e é o novo vetor para a consulta modificada, R é o conjunto de vetores dos documentos relevantes e S é o conjunto de vetores dos documentos não-relevantes. O conjunto R é usado em situações em que o usuário provê retroalimentação positiva (indica quais são relevantes). Já o conjunto S é usado em situações em que o usuário provê retroalimentação negativa (indicando quais documentos são não-relevantes). A importância de cada tipo de julgamento, bem como da consulta original, pode ser controlada pelos pesos α, β e γ, determinados experimentalmente. Se o sistema prover apenas retroalimentação positiva, então pode-se fazer $\gamma = 0$ ou também $S = D - R$, sendo D o conjunto total de documentos na coleção. Ajuste análogo pode ser realizado para quando o sistema provê apenas retroalimentação negativa.

O conjunto R pode ser utilizado em sua totalidade ou em parte. A quantidade ideal de elementos de R a participar do algoritmo é determinada empiricamente, pois depende das características da consulta e do corpus de documentos. Outro fator determinado empiricamente é a quantidade de termos a considerar para cada vetor de R [White et al. 2006].

O novo vetor da consulta é o vetor original acrescido de termos que aumentem o poder de discriminação da consulta. Sendo assim, a consulta modificada contém novos termos, obtidos dos documentos relevantes, bem como os pesos dos termos pré-existentes são ajustados. Se, durante a execução do algoritmo, algum termo atinge um valor nulo ou negativo, o termo é removido da consulta.

4.2 Considerações finais

Este capítulo expôs o problema de modificação de consulta, como solução para minimizar a divergência de vocabulário em sistemas de RI. Nessa direção, foram introduzidos brevemente os principais métodos de expansão, com ênfase nos métodos de retroalimentação de relevância. Na exposição de retroalimentação de relevância foram apresentados os comportamentos das três modalidades dessa técnica e, complementarmente, trabalhos relacionados que as empregam.

5 *Abordagem RISC-RIR para recuperação de informações sensível ao contexto usando retroalimentação implícita de relevância*

Neste capítulo apresenta-se a abordagem RISC-RIR desenvolvida para integrar o contexto de trabalho dos usuários em um mecanismo para retroalimentação implícita de relevância, de forma a prover recuperação personalizada de informação. Usando Raciocínio Baseado em Casos (RBC) para selecionar as evidências para retroalimentação, a abordagem define uma arquitetura que é capaz de gerenciar funcionalidades requeridas para processamento de contexto durante os ciclos de RBC, provendo evidências para expandir consultas com base no contexto de trabalho dos usuários.

O restante deste capítulo está organizado da seguinte forma: a seção 5.1 apresenta os repositórios de dados utilizados na abordagem, o que inclui a apresentação da ontologia de contexto e suas formas de instanciação e consulta; e a seção 5.2 detalha a arquitetura da abordagem, o que envolve a coleta e raciocínio de dados de contexto bem como o método de expansão de consulta.

5.1 Repositórios da abordagem RISC-RIR

De forma a gerenciar as informações necessárias à abordagem, são definidos dois repositórios: o repositório de documentos, que armazena os documentos e os ínidices; e o repositório de contexto, que armazena a ontologia de contexto e suas instâncias, as quais são obtidas por meio da transformação dos registros de interação do usuário com o ambiente eletrônico de trabalho. Esses repositórios são detalhados a seguir.

5.1.1 Repositório de documentos

De modo a prover acesso eficiente aos documentos durante a expansão e o processamento de consulta, foi definido um repositório que armazena os documentos e as estruturas de indexação dos conteúdos. A figura 5.1 apresenta uma visão geral do repositório de documentos.

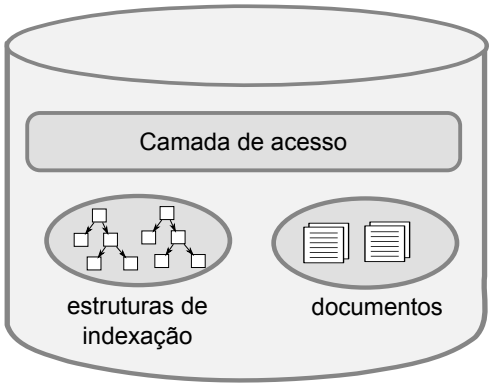

Figura 5.1: Repositório de documentos

Conforme pode ser notado pela figura 5.1, o repositório de documentos armazena, além dos documentos em si, também as estruturas de indexação dos conteúdos e dos documentos. Além disso, tanto o acesso aos documentos quanto às estruturas de indexação é realizado por meio de uma camada de acesso, que consiste em uma API (*Application Programming Interface*) para ser acessada pelos outros módulos da arquitetura da abordagem (mais detalhes sobre tais módulos são fornecidos na seção 5.2). Para indexar os documentos, é utilizado o modelo de espaço de vetores e esquema TF-IDF para definição dos pesos dos termos (conforme especificado na seção 2.2, página 20). A próxima seção apresenta o repositório de contexto.

5.1.2 Repositório de contexto

Para modelar as situações dos usuários enquanto interagem com o sistema de RI, foi construída uma ontologia de contexto em OWL. O repositório de contexto armazena a ontologia de contexto e suas instâncias. De modo a habilitar o armazenamento e a consulta de forma eficiente, o repositório de contexto é um repositório de triplas (*triplestore*[1]) compatível com SPARQL[2] (linguagem normativa do W3C para consulta

[1] Denomina-se repositório de triplas (*triplestore*) qualquer sistema gerenciador de dados com funcionalidades para armazenamento e consulta de dados em RDF, em nível de triplas. O termo *triple* remete ao modelo de dados baseado em grafos do RDF, constituído por triplas no formato $< s, p, o >$

[2] `http://www.w3.org/TR/rdf-sparql-query/`

em dados RDF). A seguir será apresentado o modelo da ontologia de contexto, assim como exemplos de instanciação e consulta dessa ontologia.

A ontologia de contexto

A ontologia de contexto representa atributos contextuais que são relevantes para caracterizar a situação do usuário no ambiente eletrônico de trabalho enquanto busca informação. Para subsidiar a criação da ontologia, foi conduzida uma revisão bibliográfica de modelos de contexto (mais detalhes sobre estes modelos na seção 3.1.2, página 40). A partir dos modelos revisados foram especificados os requisitos e definidos os principais conceitos e propriedades que a ontologia deveria apresentar.

A ontologia de contexto resultante organiza uma situação em perfis, sendo que cada perfil representa uma dimensão de contexto relevante em ambientes interativos para acesso a informações. A figura 5.2 fornece uma visão geral da organização da ontologia.

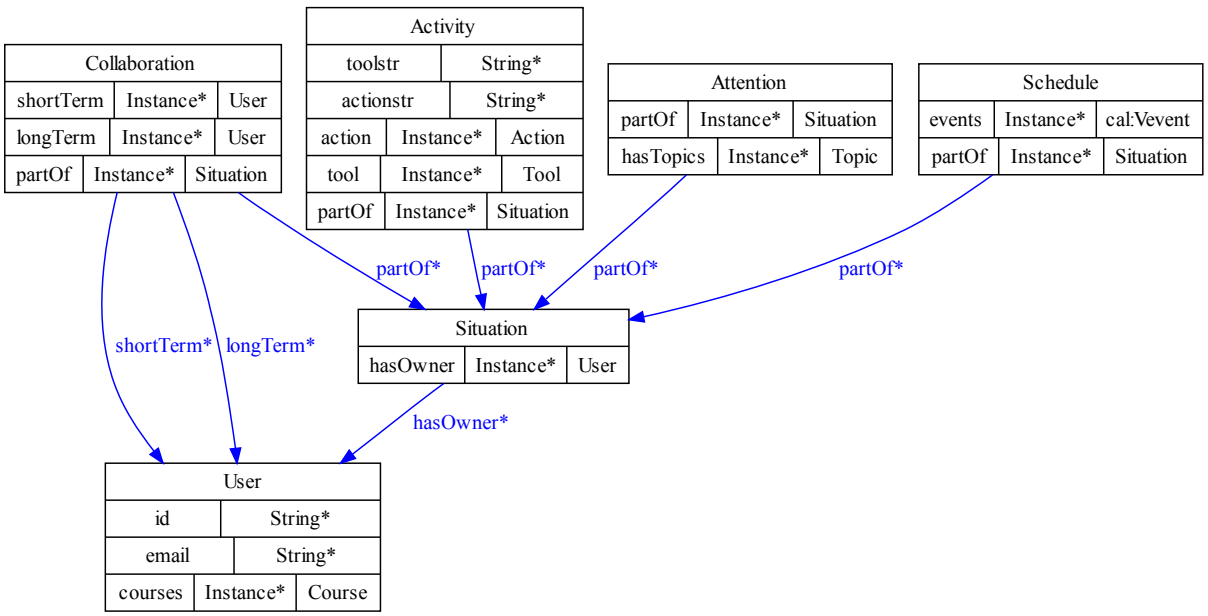

Figura 5.2: Principais conceitos da ontologia de contexto

Conforme pode ser notado pela figura 5.2, os seguintes perfis englobam atributos recorrentes que podem ser usados para caracterizar a situação do usuário:

- ***User*:** perfil de usuário, armazena informações para caracterização do usuário, como nome, informações de contato e cursos dos quais participa;

- ***Schedule*:** perfil de compromissos, descreve os compromissos que o usuário de-

verá atender ao longo de seu fluxo de trabalho, o que provê ciência de qual tarefa o usuário deveria estar participando no momento em que está buscando informação;

- ***Activity*:** perfil de atividade, caracteriza quais recursos o usuário está explorando quando dispara uma consulta, como discussão em fóruns, redação de relatórios, leitura de notas de aulas, interação em um *chat*, e assim por diante;

- ***Attention*:** perfil de atenção, especifica quais tópicos o usuário dedicou atenção recentemente de forma a caracterizar seus padrões de consumo de informação a curto prazo;

- ***Collaboration***: perfil de colaboração, lista os colaboradores do usuário de forma a descrever seu contexto social. Colaboradores podem ocorrer em interações de curto prazo (como em conversações em comunicadores instantâneos) ou interações de longo prazo (como grupos de trabalho).

Alguns dados dos perfis tem origem estática, como os dados do perfil de usuário. Outros dados tem origem dinâmica, como os perfis de atividade e atenção, os quais poderão sofrer variações durante a interação do usuário com o sistema. Os meios pelos quais os dados são obtidos, e como esses requisitos serão tratados, será discutida em mais detalhes na seção 5.2.1. A seguir são ilustradas a instanciação e a consulta da ontologia de contexto.

Instanciação e consulta dos dados de contexto

De forma a ilustrar a instanciação de situações modeladas na ontologia de contexto, formulou-se um cenário de exemplo no domínio de aprendizado ubíquo de Medicina. Esse cenário é baseado em requisitos obtidos durante o desenvolvimento de um dos estudos de caso da pesquisa. Em particular, o cenário apresentado a seguir é baseado no estudo de caso do Portfólio Reflexivo Eletrônico (mais detalhes sobre esse estudo de caso são providos no capítulo 6, seção 6.1).

Cenário : Ana é uma estudante de Medicina cujas atividades de aprendizado são assistidas por um ambiente de aprendizado ubíquo. O grupo de aprendizado de Ana está visitando residências de pacientes num pequeno distrito fora da universidade, investigando como moradores doentes estão conduzindo seus tratamentos. Um paciente informa a Ana que recentemente ele foi diagnosticado como portador do vírus da hepatite C. Subitamente, Ana recorda que ela e seu grupo de

aprendizado estavam analisando descrições sucintas de procedimentos para tratamento de hepatite em sessões de aprendizado anteriores e que tornar-se ciente desses documentos poderia enriquecer os resultados da entrevista. Ana então decide buscar tais documentos no ambiente de aprendizado ubíquo usando seu telefone celular. Usando o limitado teclado numérico do telefone, Ana dispara uma consulta curta "procedimentos hepatite" em seu portfólio eletrônico.

Quando Ana submete uma consulta ao ambiente, a arquitetura já terá instanciado perfis descrevendo os atributos de seu contexto de trabalho compondo uma situação. Um exemplo de situação para o cenário descrito é ilustrado na Figura 5.3, usando a notação sintática Turtle[3]:

```
:std_4
  a foaf:Person ;
  foaf:name "Ana" ;
  :isOwner :ctx_prof_10 .
:ctx_prof_10
  a :Situation ;
  :comprises (:sched_prof_3 :act_prof_10
        :collab_prof_35  :att_prof_48) .
:sched_prof_3
  a :Schedule ;
  :cEventPlain "C4 - Prática Profissional" .
:act_prof_10
  a :Activity ;
  :tool :EletronicPortfolio ;
  :action :Browsing .
:collab_prof_35
  a :Collaboration ;
  :longTerm (:std_7 :std_50 :std_43) .
:att_prof_48
  a :Attention ;
  :topics ("virus" "figado" "hepatologia") .
```

Figura 5.3: Exemplo de situação

O exemplo de situação da Figura 5.3 descreve o compromisso programado atualmente para Ana (Prática Profissional), sua atividade corrente (estava navegando seu portfólio eletrônico quando a consulta foi disparada), e lista como seus colaboradores alguns estudantes membros de seu grupo de trabalho. Adicionalmente, lista os tópicos principais aos quais Ana dedicava atenção enquanto navegava seu portfólio (`virus`, `figado`, `hepatologia`). Contando com esta representação de situação, a arquitetura da abordagem pode raciocinar sobre os perfis armazenados no repositório de contexto, obter evidências, e expandir a consulta de Ana para abranger seu contexto de trabalho.

[3] `http://www.dajobe.org/2004/01/turtle/`

Quando são armazenadas, as situações permanecem no repositório de contexto. Para habilitar armazenamento e consulta de forma eficiente, o repositório de contexto é um repositório de triplas (*triplestore*) compatível com SPARQL. Cada situação no repositório de contexto é um grafo nomeado que é associado a um ou mais documentos.

Há várias abordagens para se trabalhar com grafos nomeados em repositórios de triplas (conforme listado na seção 3.1.2). Neste trabalho adotou-se o recurso de RDF *datasets* provido pela especificação da linguagem SPARQL, que é compatível com a maioria dos repositórios de triplas. Um *dataset* RDF é um conjunto formado por um grafo não-nomeado, chamado grafo implícito, e zero ou mais grafos nomeados. A figura 5.4 ilustra por meio de um exemplo simples, usando a notação Turtle, como grafos nomeados são usados para associar situações e documentos no repositório de contexto.

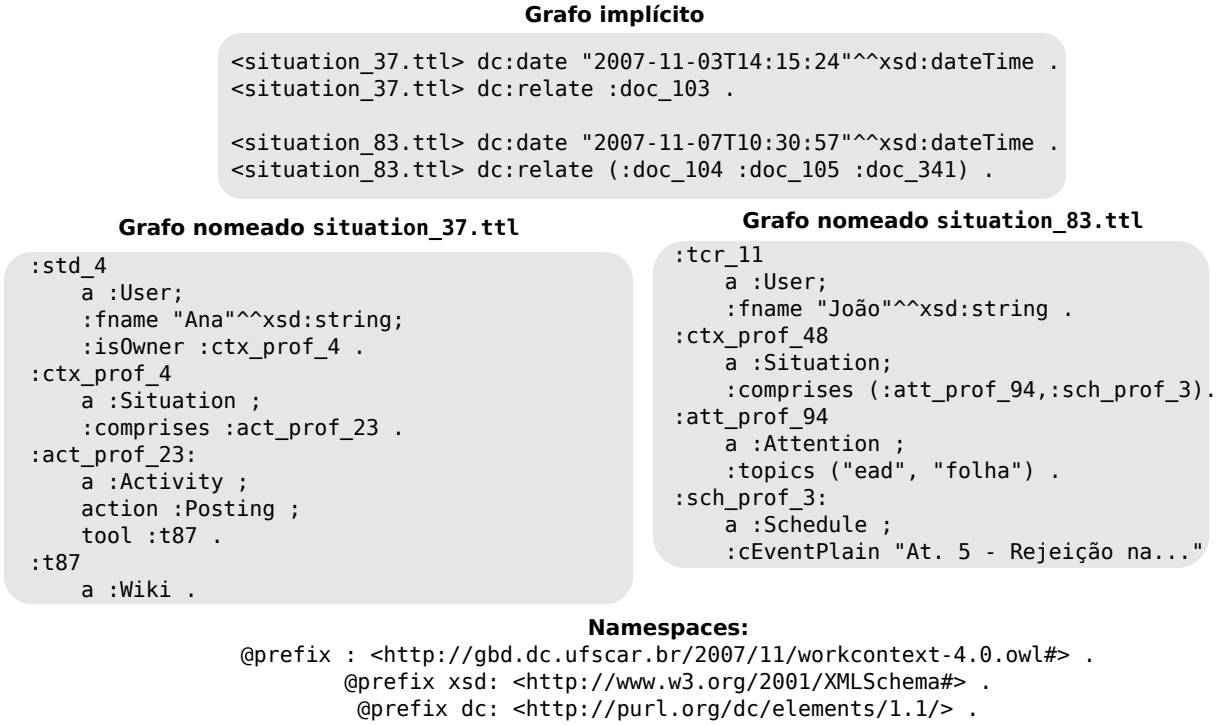

Figura 5.4: Visão geral da estratégia de modelagem de situações usando grafos nomeados

Na figura 5.4 o grafo nomeado `situation_37.ttl` descreve a situação do usuário `std_4`. A situação desse usuário é formada por um perfil de atividade (`Activity`) composto por uma ação de escrita (`Posting`) em uma ferramenta de autoria do tipo *wiki*. Já o grafo nomeado `situation_83.ttl` descreve a situação do usuário `tcr_11`, composta por um perfil de atenção (`Attention`) e por um perfil de compromissos (`Schedule`).

O grafo implícito é responsável por registrar as propriedades dos grafos nomeados. Sendo assim, este grafo mantém uma propriedade `dc:date` de forma a associar um atributo temporal a cada situação. Além disso, o grafo implícito também registra as associações entre situações e documentos. Na figura 5.4, a situação `situation_37.ttl` está associada a apenas um documento (`doc_103`). Já a situação `situation_83.ttl` está associada a três documentos: `doc_104, doc_105, doc_341`.

Por meio desta estratégia de modelagem de situações, a arquitetura da abordagem pode obter as evidências para retroalimentação usando consultas em SPARQL. A figura 5.5 apresenta um exemplo de consulta para obter evidências.

```
SELECT DISTINCT ?doc
WHERE {

  ?g dc:relate ?doc .
  ?g dc:date . FILTER (?date >
       "2007-11-03T00:00:00"^^xsd:dateTime)

  GRAPH ?g {
  :std_4 :isOwner ?ctx .
     ?ctx :comprises ?act .
     ?act :tool ?t .
     ?t a :Wiki .
  }
}
ORDER BY DESC ?date .
LIMIT 20
```

Figura 5.5: Exemplo de consulta SPARQL para obter evidências

A consulta da figura 5.5 irá recuperar um conjunto ordenado de 20 identificadores de documentos. Serão recuperados os identificadores associados a situações criadas a partir de uma data pré-definida. Restringe-se as situações envolvidas apenas àquelas que pertencem ao usuário `std4` e que foram registradas quando o usuário estava interagindo com uma wiki.

Em suma, a ontologia de contexto provê as bases para representar as situações processadas durante a expansão de consulta, e o repositório de contexto armazena a ontologia de contexto e suas instâncias. A representação das situações como grafos nomeados em RDF permite associá-las aos documentos, o que habilita a recuperação de evidências usando consultas em SPARQL. O repositório de contexto é utilizado pela arquitetura da abordagem, a qual será tratada na próxima seção.

5.2 Arquitetura da abordagem RISC-RIR

A arquitetura desenvolvida para a abordagem RISC-RIR trabalha com uma memória de contexto, mantida via Raciocínio Baseado em Casos. A memória de contexto é constituída de situações baseadas em ontologia (grafos nomeados), armazenados no repositório de contexto, e é usada para refinar as consultas dos usuários. As instâncias do modelo de contexto permitem encontrar quais documentos são úteis para o usuário durante a busca, comparando a situação atual com situações anteriores, e mais tarde aplicando estes documentos como evidências para prover retroalimentação de relevância. Uma visão geral dos principais módulos dessa arquitetura e suas interações é provida na figura 5.6.

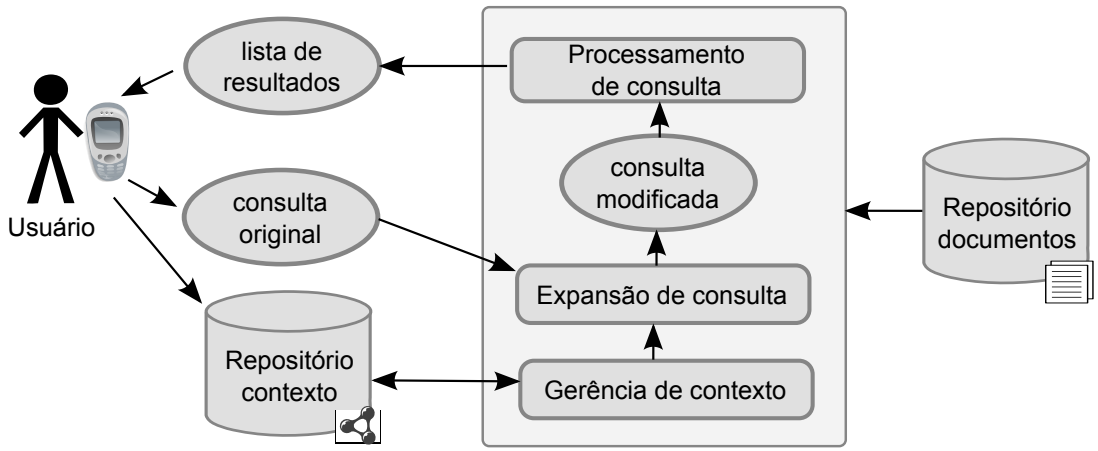

Figura 5.6: Arquitetura da abordagem RISC-RIR

A funcionalidade central da arquitetura agrega a gerência de contexto e a expansão de consulta, atendidas pelos módulos *Gerência de contexto* e *Expansão de consulta*, respectivamente. O módulo *Gerência de contexto* tem um papel importante na arquitetura pois apresenta funcionalidades para capturar, processar e armazenar os dados de contexto. Todas essas funcionalidades têm o propósito de subsidiar o provimento de evidências ao módulo *Expansão de consulta*.

Além disso, a maioria das operações realizadas pelo módulo *Gerência de contexto* são executadas assincronamente ao processamento da consulta, de forma a não impor sobrecarga adicional (*overhead*) no processo de RI. O módulo *Expansão de consulta*, por sua vez, é responsável por modificar a consulta original, adicionando e ajustando os pesos dos termos, por meio de retroalimentação implícita de relevância. Para isso, o módulo *Expansão de consulta* baseia-se nas evidências fornecidas pelo módulo *Gerência de contexto*. Evidências são os documentos que estão associados às situações armaze-

nadas que são similares à situação atual.

O caso de uso típico da arquitetura ocorre quando o usuário submete uma consulta ao sistema. Após receber a consulta, o módulo *Expansão de consulta* é ativado para transparentemente proceder com a expansão da consulta. Como parte deste processo, o módulo *Expansão de consulta* interroga o módulo *Gerência de contexto* em busca de evidências.

Assincronamente, o módulo *Gerência de contexto* instancia os perfis do contexto atual do usuário. Para isso, são consultados os registros de interação coletados no ambiente eletrônico de trabalho do usuário. Obtidos os perfis do contexto atual do usuário, o módulo *Gerência de contexto* consulta o repositório de contexto, comparando o atual contexto de trabalho do usuário (situação atual) com as situações que estão armazenadas no repositório de contexto. Como as situações armazenadas são associadas aos documentos correspondentes por meio do grafo implícito do repositório de contexto, será retornada uma lista de evidências (documentos) que são relacionadas a situações semelhantes à situação atual. Ao receber as evidências, o módulo *Expansão de consulta* expande a consulta usando as evidências, e encaminha a consulta expandida ao módulo de *Processamento de consulta*, que irá determinar os documentos resultantes de acordo com o modelo de espaço de vetores. O resultado deste passo é uma lista ordenada por relevância (*ranking*) de resultados que será apresentada ao usuário.

Nota-se com isso que há um estreito relacionamento entre consultas, documentos e situações na arquitetura. Em particular, uma situação constitui um grupo de perfis de contexto que foi instanciado num instante específico da interação do usuário com o sistema. As situações associadas aos documentos constituem contextos de interações passadas, enquanto a situação associada à consulta constitui o contexto atual da interação do usuário. A figura 5.7 ilustra as relações entre situações, documentos e consultas, bem como provê alguns detalhes internos de como estas relações são exploradas na arquitetura.

Como pode ser notado pela figura 5.7, cada documento armazenado no repositório de documentos (*d1,..,dn*) possui uma ou mais situações associadas (*s1,...,sm*), que permanecem armazenadas no repositório de contexto. As situações associadas aos documentos (denominadas situações passadas) descrevem o que o usuário estava fazendo quando interagiu com o documento associado em particular. Analogamente, cada consulta que o usuário submete ao sistema, também possui uma situação associada (*sa*, denominada situação atual). A partir desta configuração, o problema que

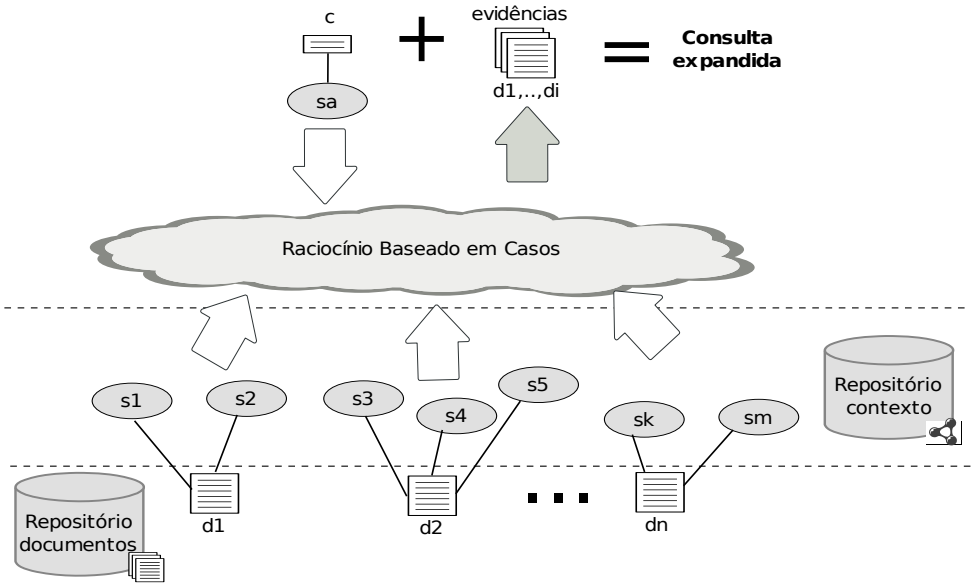

Figura 5.7: Comportamento do processo de seleção de evidências para expansão de consulta

o sistema precisa resolver consiste em encontrar situações passadas que são similares à situação corrente. Para raciocinar desta forma, é utilizado o raciocínio baseado em casos, que permite inferir as situações por analogia e similaridade. O resultado desse processo é um conjunto de documentos (evidências *d1,...,di*) que indiretamente estão relacionados à situação corrente. As evidências são então usadas no algoritmo de retroalimentação de relevância para expandir a consulta original do usuário.

A próxima seção tratará de questões acerca da gerência de dados de contexto, mais especificamente a captura, transformação e raciocínio sobre os dados de contexto.

5.2.1 Gerência de contexto

Um fator importante que a arquitetura deve tratar é como efetivamente gerenciar os dados de contexto. Com esse propósito, o módulo *Gerência de contexto* é responsável por adaptar, armazenar e raciocinar sobre os dados de contexto. Para atingir esses objetivos, esse módulo conta com alguns componentes, cujas interações são ilustradas na figura 5.8.

Como pode ser notado pela figura 5.8, os dados de contexto são adquiridos pelo componente de Coleta, que os transformará em triplas RDF para compor as situações. O componente de coleta comunica-se com a camada de acesso quando é necessário armazenar a situação atual do usuário. O componente de Raciocínio realiza o raciocínio

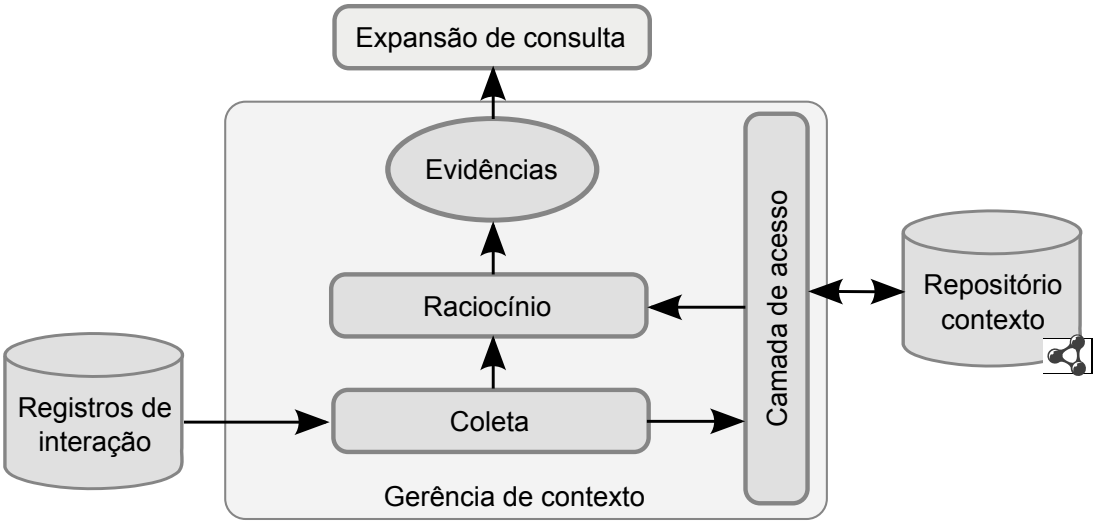

Figura 5.8: Visão ampliada do módulo de Gerência de contexto

baseado em casos, a partir da situação atual do usuário (informada pelo coletor) e das situações armazenadas (obtidas via camada de acesso). Como a situação do usuário é instanciada assincronamente ao processamento da consulta, as evidências obtidas pelo raciocínio são mantidas em um recurso de sincronização (*cache*) que é acessada pelo módulo de expansão de consulta. A seguir serão detalhados o processo de coleta de dados de contexto e o raciocínio sobre esses dados.

Coleta de dados de contexto

Os dados de contexto são adquiridos a partir da interação do usuário com o ambiente eletrônico de trabalho. Tais registros podem ser obtidos de diferentes maneiras, a depender do cenário em que a abordagem é aplicada. Por exemplo, em ambientes de aprendizado eletrônico, é natural a presença de registros de interação, que são mantidos pelo ambiente para fins de avaliação. Os principais ambientes de aprendizado eletrônico, a exemplo de Moodle[4], Sakai[5] e WebCT[6], dispõem de funcionalidades para registrar as interações dos usuários com os recursos do ambiente, em geral para finalidades avaliativas. A figura 5.9 ilustra um relatório gerado pelo ambiente Sakai a partir dos registros de interação.

Já em ambientes voltados à Web em geral, os registros de interação podem ser coletados por soluções no lado do cliente, via complementos (*plugins*) instalados no navegador ou na estação de trabalho do usuário; ou por soluções no lado do servidor,

[4] http://moodle.org/
[5] http://sakaiproject.org/portal
[6] http://www.blackboard.com/

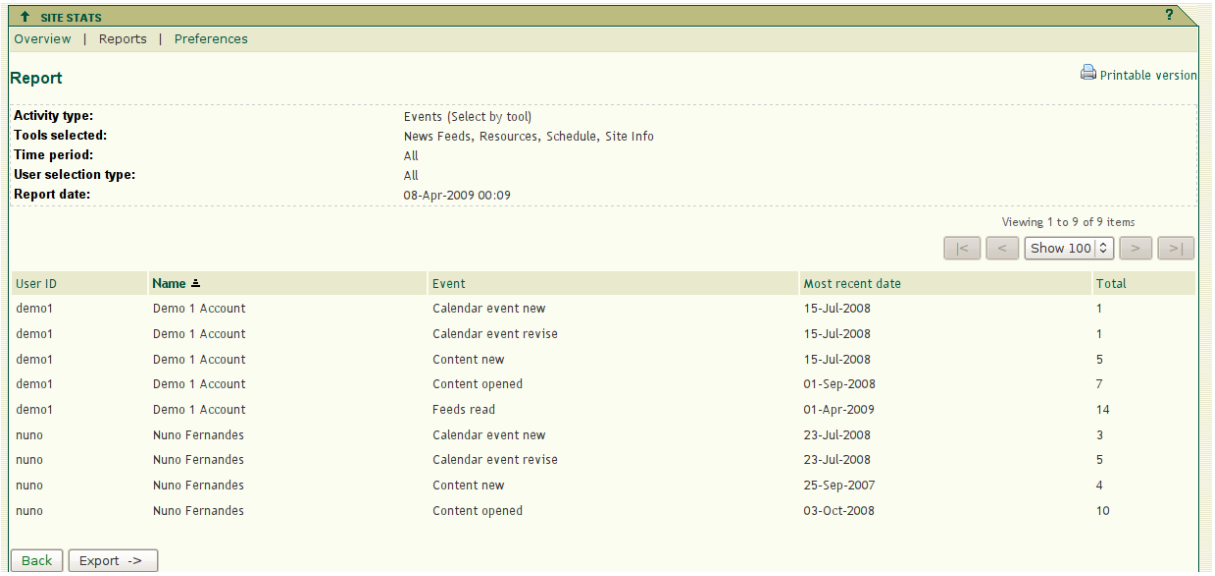

Figura 5.9: Exemplo de relatório gerado a partir dos registros de interação do ambiente Sakai

via instrumentação das aplicações com o propósito específico de reportar eventos de interação do usuário.

Independentemente do método de captura dos dados, os registros de interação são obtidos pelo componente de coleta e transformados para tornarem-se instâncias da ontologia de contexto. O processo de conversão é dependente do modelo de dados empregado para armazenar os registros, logo o coletor precisa ser especializado para cada ambiente no qual a arquitetura for integrada.

Os registros precisam estar armazenados em um banco de dados relacional e, para a conversão dos dados para o modelo da ontologia de contexto, o componente de coleta é baseado no método da ferramenta Triplify[7] [Auer et al. 2009]. Nessa ferramenta, cada conceito da ontologia é representado por um conjunto de consultas em SQL que recupera as propriedades do conceito. Os resultados das consultas são então unidos pelos identificadores e convertidos em triplas RDF.

Independentemente dessas particularidades, o componente de coleta é responsável por delimitar uma situação em meio ao fluxo de entradas no registro de interação. Para esta finalidade, periodicamente é submetida uma consulta ao banco de dados do ambiente para obter os novos registros. A manutenção das situações atuais do ambiente é realizada pelo algoritmo da figura 5.10.

Os registros de interação são extraídos num intervalo periódico k, que pode ser

[7] http://triplify.org/

```
1 for each instant k:
2       regs = getRegs(last)
3       last = regs.lastTime()
4       for each r in regs:
5             for each s in sits:
6                   if s.hasExpired():
7                         sits.remove(s)
8                   else if r.getUser() == s.getUser():
9                         s.merge(triplify(r))
10                        r = null
11            if r != null:
12                  sits.add(new sit(triplify(r)))
```

Figura 5.10: Algoritmo para coleta e transformação de registros de interação

configurado. Para reduzir a quantidade de dados a processar, a cada extração são obtidas apenas as entradas que foram criadas a partir da última extração. O algoritmo mantém uma lista *sits*, que contém todas as situações ativas no ambiente, ordenadas pela atualização mais recente. Cada entrada de registro é analisada para verificar se pode agregar-se a algumas das situações disponíveis. Se for possível, a entrada é convertida em triplas RDF e é agregada ao grafo. Se não for possível agregar a entrada a nenhuma das situações disponíveis, uma nova situação é criada. O algoritmo também verifica se alguma situação expirou e, em caso afirmativo, destrói a situação.

Desta forma, o componente de coleta mantém uma lista contendo uma situação para cada usuário ativo no ambiente. A situação de cada usuário é usada pelo componente de Raciocínio para executar o raciocínio baseado em casos. O resultado do raciocínio é uma lista de evidências (identificadores de documentos), que serão usados pelo módulo de Expansão de consulta para expandir a consulta do usuário.

A seguir, será detalhado o processo de seleção de evidências pelo componente de Raciocínio.

Raciocínio sobre os dados de contexto

Para selecionar as evidências que serão usadas na expansão de consulta, o componente Raciocinador do módulo Gerente de contexto opera com raciocínio baseado em casos. Para aplicar RBC no problema de seleção de evidências para retroalimentação implícita de relevância, foi necessário adaptar a metodologia RBC para as particularidades deste problema. A primeira decisão de projeto foi definir como representar os casos e a base de casos.

Em RBC, um caso é composto por um problema e sua respectiva solução. Instanciando a definição para este projeto, o problema é representado pela situação do usuário

e a solução é representada pelo conjunto de documentos associados à situação. Logo, um caso é um par (s,D) tal que s é uma situação e D é um conjunto de documentos associados à situação s. A representação da situação atual do usuário é um caso especial dessa definição, em que s é a situação atual do usuário e D é um conjunto unitário formado pela consulta original do usuário.

Nota-se que a representação do caso é dependente da garantia de haver associações entre situações e documentos. Em vista da modelagem adotada de representar situações como grafos nomeados, e associações entre situações e documentos como propriedades no grafo implícito, o repositório de contexto pode naturalmente operar como uma base de casos.

A partir destas definições, foi necessário definir como um ciclo de RBC pode ser integrado às tarefas de retroalimentação implícita de relevância. A integração é apresentada na figura 5.11.

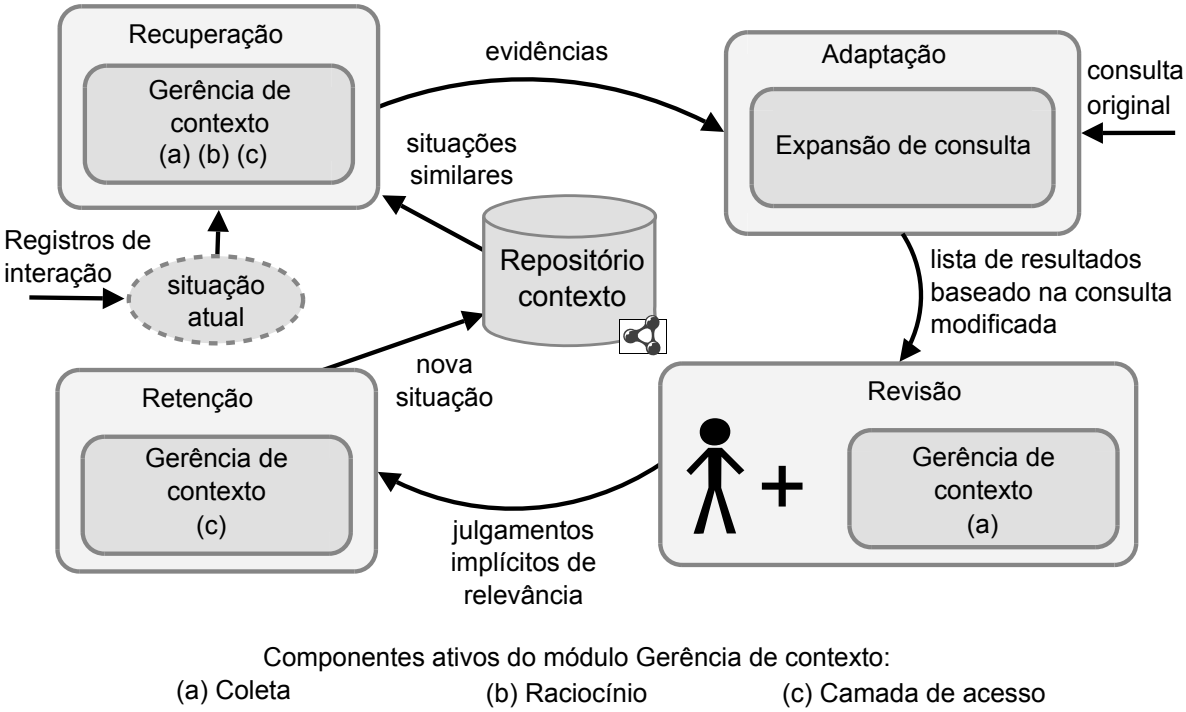

Figura 5.11: Ciclo de Raciocínio Baseado em Casos adaptado para Retroalimentação Implícita de Relevância

Os quatro passos (recuperação, adaptação, revisão, retenção) de um ciclo RBC, são satisfeitos na arquitetura da seguinte forma:

(1) Recuperação: O componente de Coleta do módulo de Gerência de contexto constrói a situação atual do usuário que é usada pelo componente de Raciocínio para

obter situações similares. Como resultado, o Repositório de contexto fornece uma lista de identificadores de documentos que foram usados em situações que coincidem em certo grau com a situação atual do usuário. Esses identificadores de documentos serão usados como entrada para o algoritmo de retroalimentação de relevância;

(2) Adaptação: a adaptação da solução é realizada pelo módulo de Expansão de consulta, que adapta a consulta original do usuário a partir da lista de evidências;

(3) Revisão: os resultados para a consulta modificada são apresentados ao usuário. Se o usuário clicar em um documento na lista, é denotado que o documento foi útil à consulta, implicando em julgamento implícito de relevância por parte do usuário; este comportamento é obtido dos registros de interação e é detectado pelo componente de Coleta;

(4) Retenção: se o módulo de Gerência de contexto detecta um julgamento implícito de relevância, a situação atual do usuário é armazenada no Repositório de contexto, em conjunto com uma associação ao documento correspondente. Para os próximos documentos julgados sob a mesma situação, armazena-se uma nova associação para cada documento. Consequentemente, o documento será memorizado com a situação e posteriormente poderá participar de outros ciclos de raciocínio.

No passo de recuperação, é necessário obter situações armazenadas no Repositório de contexto para computar similaridade entre a situação atual e as situações armazenadas. Uma alternativa adotada de início foi a utilização de consultas SPARQL de similaridade, usando a extensão iSPARQL [Kiefer et al. 2008], que permite a definição de estratégias de similaridade diretamente na consulta. A vantagem principal dessa estratégia é a transferência do custo do cálculo de similaridade para o repositório de triplas. Como desvantagens, aponta-se o aumento da complexidade das consultas e a necessidade de extensão do processador de consulta do repositório de triplas, o que diminui a reusabilidade da solução.

Como alternativa, adotou-se a estratégia de recuperar as situações considerando dois estágios: (1) filtragem estrutural e (2) similaridade de atributos. No estágio de filtragem estrutural, são selecionadas apenas as situações armazenadas que coincidem estruturalmente com a situação atual, dessa forma diminuindo o espaço de busca do cômputo de similaridade; no estágio de similaridade de atributos, as situações resultantes do passo 1 são comparadas com a situação atual de acordo com as métricas de similaridade adotadas, na memória principal. Em vista do cálculo de similaridade ser um processo custoso, a estratégia de dois estágios permite filtrar o conjunto de situa-

ções que participarão do cálculo.

O estágio de filtragem estrutural consiste na conversão da situação atual do usuário em uma consulta SPARQL que permita recuperar do repositório de contexto apenas as situações com a mesma estrutura da situação atual. Para isso, aplica-se um algoritmo de transformação na situação atual, apresentado na figura 5.12.

```
1   for each triple in sit:
2       (s,p,o) = triple.split()
3       if masks.hasMapping(s):
4           s = masks.getMapping(s)
5       else:
6           s = masks.newMapping(s)
7       if masks.hasMapping(o):
8           o = masks.getMapping(o)
9       else:
10          o = masks.newMapping(o)
11      patterns.add(new pattern(s,p,o))
12  build_query(patterns)
```

Figura 5.12: Algoritmo para converter situação atual do usuário em consulta SPARQL

O algoritmo da figura 5.12 irá processar cada tripla da situação atual. Para cada tripla, os componentes sujeito e objeto da tripla decomposta sofrem substituições de variáveis, isto é, os literais e instâncias são consistentemente substituídos por variáveis. Após o processamento de todas as triplas, é construída a consulta usando as triplas transformadas. Um exemplo de execução do algoritmo é apresentado na figura 5.13.

Na figura 5.13 é apresentada a situação a ser transformada na sintaxe Turtle simplificada e, para efeito de clareza, a mesma situação na forma expandida, que torna mais evidente a delimitação das triplas. O algoritmo irá decompor cada tripla e substituir os componentes sujeito e predicado, que sejam instâncias ou literais, por variáveis. As triplas com variáveis substituídas são usadas como padrões de casamento na consulta SPARQL, que recupera os grafos que casam com os padrões.

Adicionalmente, o algoritmo de transformação pode ser parametrizado para incluir condições adicionais de seleção como restrições temporais e limiar superior do número de situações recuperadas. Com estas medidas, o estágio de filtragem estrutural permite filtrar o conjunto de situações apenas àquelas que são super-grafos da situação do ponto de vista estrutural, com arestas coincidentes, bem como atendendo a restrições adicionais de seleção caso o espaço de situações seja muito extenso.

O resultado do estágio de filtragem estrutural é uma lista de grafos nomeados re-

Situação na forma simplificada

```
:tcr_11
    a :User;
    :fname "João"^^xsd:string ;
    :isOwner :ctx_prof_48 .
:ctx_prof_48
    a :Situation;
    :comprises (:att_prof_94,:sch_prof_3).
:att_prof_94
    a :Attention ;
    :topics ("ead", "folha") .
:sch_prof_3:
    a :Schedule ;
    :cEventPlain "At. 5 - Rejeição na..."
```

Situação na forma expandida

```
:tcr_11      a            :User .
:tcr_11      :fname       "João"^^xsd:string .
:tcr_11      :isOwner     :ctx_prof_48 .
:ctx_prof_48 a            :Situation
:ctx_prof_48 :comprises   :att_prof_94 .
:ctx_prof_48 :comprises   :sch_prof_3 .
:att_prof_94 a            :Attention .
:att_prof_94 :topics      "ead" .
:att_prof_94 :topics      "folha" .
:sch_prof_3  a            :Schedule .
:sch_prof_3  :cEventPlain "At. 5 - Rejeição na..." .
```

Consulta resultante da transformação

```
SELECT DISTINCT ?g
WHERE {
  GRAPH ?g {
  :v0     a            :User .
  :v0     :fname       ?v1 .
  :v0     :isOwner     ?v2 .
  ?v2     a            :Situation
  ?v2     :comprises   ?v3 .
  ?v2     :comprises   ?v4
  ?v3     a            :Attention .
  ?v3     :topics      ?v6 .
  ?v3     :topics      ?v7 .
  ?v4     a            :Schedule .
  ?v4     :cEventPlain ?v8 .
  }
}
```

Figura 5.13: Transformação de situação em consulta SPARQL

presentando as situações selecionadas. Estas situações participam do próximo estágio, que envolve a determinação da similaridade da situação atual em relação às situações selecionadas, com base nos atributos das situações.

Para o estágio de similaridade de atributos foram reusadas as funcionalidades para cálculo de similaridade do *framework* jColibri. Este *framework* para RBC calcula a similaridade entre os casos utilizando o algoritmo k-NN (*k-Nearest Neighbors*). Por este algoritmo, dado um caso usado como consulta, aplica-se uma estratégia de similaridade com cada caso armazenado e apenas os k casos com maior grau de similaridade são selecionados. A estratégia de similaridade é composta por funções de similaridade. As funções de similaridade podem ser locais (aplicadas sobre um atributo simples) ou funções de similaridades globais (aplicadas sobre atributos compostos; um atributo composto pode ser formado por um número arbitrário de atributos, até mesmo o caso completo).

O *framework* ainda especializa as estratégias para casos baseados em ontologias. Para os casos neste formato, uma estratégia de similaridade pode ser definida mapeando cada tripla da ontologia a uma função de similaridade local. A similaridade global é determinada por uma função de combinação que sumariza as similaridades

locais. Na abordagem RISC-RIR, o componente de Raciocínio configura a estratégia de similaridade para utilizar soma ponderada como função de combinação:

$Sim(s,p) = \sum_{i=1}^{n} w_i \times sim(s_i, p_i)$

Na função de combinação, s e p são situações, w_i é o peso atribuído à tripla i e $sim(s_i, p_i)$ é a função local de similaridade das situações s e p com referência à tripla i. O resultado da função de combinação é um valor real que sumariza as funções de similaridade locais. Tanto as funções locais quanto os pesos são configuráveis como parâmetros da arquitetura. Estes parâmetros são configuráveis pois, dependendo do modelo de dados dos registros de interação de cada ambiente de aprendizado, apenas um subconjunto da ontologia de contexto será utilizada. Além disso, é fornecida flexibilidade para definir quais atributos serão mais importantes no processo de similaridade.

O efeito do estágio de similaridade de atributos é uma lista de situações armazenadas que são similares à situação atual do usuário. Em posse dessas situações, o componente de Raciocínio recupera a partir do grafo implícito os identificadores dos documentos associados a estas situações. A quantidade de identificadores recuperados é configurável, visto que o limiar superior de evidências a alimentar o algoritmo de retroalimentação de relevância é empírico e dependente das características dos documentos indexados. A lista resultante de identificadores é utilizada pelo módulo de Expansão de consulta.

Em suma, o processo de seleção de evidências consiste na aplicação da metodologia RBC para o problema de retroalimentação implícita de relevância. Como parte deste processo, a arquitetura transforma a situação do usuário em consultas SPARQL e aplica uma estratégia de dois níveis (filtragem estrutural e similaridade de atributos) para efetivamente selecionar as evidências. A próxima seção detalha como as evidências selecionadas são aproveitadas para expandir consultas.

5.2.2 Expansão e processamento de consulta

Consultas e documentos são representados usando o modelo de espaço de vetores. Como tal, documentos e consultas são representações na forma de vetores de pesos para os termos dos documentos. Os pesos são obtidos de acordo com o esquema de pesos TF-IDF. As evidências selecionadas pelo módulo de Gerência de contexto são disponibilizadas no recurso de sincronização (*cache*) de evidências para serem acessa-

das pelo módulo de Expansão de consulta.

As evidências indicadas consistem em um conjunto $d_1, d_2, ..., d_n$ de identificadores de documentos indexados que serão usados para alimentar o algoritmo de retroalimentação de relevância. O vetor correspondente a cada evidência é obtido a partir das estruturas de indexação disponíveis no Repositório de documentos. O conjunto formado pelos vetores de evidências é representado por $R = \{\vec{d}_1, \vec{d}_2, ..., \vec{d}_n\}$ e é usado para alimentar o algoritmo de retroalimentação de relevância, representado pelo seguinte fórmula:

$$q_e = q_0 + \frac{1}{|R|} \sum_{i=1}^{n} \vec{d}_i$$

Este algoritmo é a especialização do algoritmo de Rocchio (apresentado na seção 4.1) para prover retroalimentação positiva, com $\alpha = \beta = 1$ e $\gamma = 0$. O algoritmo irá partir do vetor da consulta inicial q_0 e iterativamente aproximá-lo do centróide de documentos relevantes.

Após a expansão da consulta, a consulta q_e é executada pelo módulo de Processamento de consulta, que irá recuperar do Repositório de documentos usando a similaridade cosseno sobre os vetores da consulta e dos documentos, conforme a formulação a seguir:

$$R(d_j, q_e) = \frac{d_j \bullet q_e}{|d_j||q_e|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q_e}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q_e}^2}}$$

Esta similaridade fornece uma ordenação (*ranking*) $R(d_j, q_e) \in [0, 1]$ para cada casamento entre um vetor de documento d_j no Repositório de documentos em relação a um vetor de consulta q_e, computando o cosseno do ângulo entre os dois vetores num espaço t-dimensional, onde t é o número de termos (vocabulário) indexado pelo sistema. Uma vez que essa medida de ordenação é obtida, os resultados são formatados e apresentados ao usuário para inspeção.

5.3 Considerações finais

Este capítulo apresentou a abordagem RISC-RIR, desenvolvida para prover recuperação de informações sensível ao contexto, usando retroalimentação implícita de relevância. Nesse ínterim, foram apresentados os repositórios e a arquitetura da abordagem. Foi exposto como os perfis de contexto são representados e gerenciados, bem como os meios pelos quais as evidências para expansão de consulta são obtidas. Por

fim, demonstrou-se como a consulta é expandida usando o algoritmo de Rocchio e processada usando o modelo de espaço de vetores.

A próxima seção apresenta os estudos de caso desenvolvidos para validar a abordagem.

6 *Estudos de Caso*

Para avaliar a abordagem RISC-RIR, foram desenvolvidos dois estudos de caso em ambientes de aprendizado eletrônico. O primeiro estudo de caso foi conduzido em um ambiente para aprendizado de medicina baseado em problemas, empregado em grupos-piloto do curso de Medicina da UFSCar. O segundo estudo de caso foi conduzido em dados do ambiente Moodle utilizado pelos cursos de educação à distância da UFSCar (UAB-UFSCar).

Em ambos os estudos de caso, foi utilizado um protótipo da arquitetura implementado majoritariamente em Java. Para a manipulação da ontologia de contexto foi utilizado o framework Jena[1], que inclui uma extensão para realizar consultas em SPARQL, denominada ARQ[2]. O repositório de contexto foi instanciado com os repositórios de triplas SDB[3] (para o estudo de caso da seção 6.1) e Virtuoso[4] *Open Source Edition* (para o estudo de caso da seção 6.2). Para implementar as funcionalidades de raciocínio baseado em casos foi utilizado o framework jColibri[5]. As funcionalidades de RI (indexação de documentos e processamento de consulta) foram reusadas do *toolkit* Lemur[6] que, embora seja implementado em C++, provê interfaces de programação em Java via a biblioteca de integração SWIG[7]. Foi necessário desenvolver mapeamentos SWIG adicionais para o Lemur, pois as funcionalidades de expansão de consulta não estavam disponíveis originalmente nas interfaces em Java.

O restante deste capítulo está organizado da seguinte forma: a seção 6.1 apresenta o estudo de caso realizado sobre os dados do ambiente PRE Ubíquo (Portfólio Reflexivo Eletrônico Ubíquo) e a seção 6.2 apresenta o estudo de caso realizado sobre os dados da UAB-UFSCar (Universidade Aberta do Brasil - UFSCar).

[1] `http://jena.sourceforge.net/`
[2] `http://jena.sourceforge.net/ARQ/`
[3] `http://jena.sourceforge.net/SDB/`
[4] `http://virtuoso.openlinksw.com/wiki/main/Main/`
[5] `http://gaia.fdi.ucm.es/projects/jcolibri/`
[6] `http://www.lemurproject.org/`
[7] `http://www.swig.org/`

6.1 Estudo de caso: PRE (Portfólio Reflexivo Eletrônico) Ubíquo em Medicina

No ensino de Medicina, sobretudo nos cursos onde o processo de aprendizagem é baseado em Aprendizado Baseado em Problemas (*Problem Based Learning* ou PBL), é comum o uso de Portfólio Reflexivo (PR), um instrumento de registro de atividades do estudante que permite a reflexão do aluno quanto à sua evolução na construção de conhecimento. O conteúdo desse recurso educacional é fracamente estruturado, composto principalmente por coleções de documentos expressos em linguagem natural, o que o faz fortemente dependente de funcionalidades para recuperação de informações.

Hoje a mídia mais empregada nos PRs é o papel, sendo que alguns trabalhos, relativos a Portfólio Reflexivo Eletrônico (PRE) de propósito geral, são encontrados na literatura. A fim de que um portfólio eletrônico possa também ser acessado através de dispositivos móveis (e.g., *tablets*, *personal digital assistants*, *smartphones*) com capacidades limitadas (e.g., energia, tela, memória, processamento), o Grupo de Computação Ubíqua (GCU[8]) do DC-UFSCar tem empregado esforços no desenvolvimento de um PRE Ubíquo, cujo projeto é descrito em [Perlin et al. 2007, Santos et al. 2008].

O PRE Ubíquo objetiva permitir aos estudantes de Medicina da UFSCar usar esta ferramenta também durante as atividades práticas, que muitas vezes ocorrem fora do campus da universidade. Complementarmente, o PRE Ubíquo oferece aos estudantes ferramentas de comunicação e produtividade tipicamente disponíveis em ambientes de CSCL, como autoria colaborativa de documentos, comunicadores instantâneos, fóruns, entre outras. Além disso, visando aos requisitos de ubiquidade, o PRE Ubíquo destaca-se por apresentar funcionalidades para apresentação adaptativa dos conteúdos de acordo com as características do contexto de entrega, descritas em [Santana et al. 2007c, Santana et al. 2007a], o que permite acessar conteúdo adaptado aos recursos dos dispositivos móveis. A figura 6.1 apresenta o PRE Ubíquo acessado em duas modalidades: *desktop* e *mobile*.

De forma a avaliar os diferentes projetos de pesquisa associados ao PRE Ubíquo, diversos grupos-piloto, compostos por estudantes e professores do curso de Medicina da UFSCar, têm usado o ambiente durante as atividades educacionais. Este estudo de caso baseou-se nos registros de interação coletados durante um período de 10 semanas de utilização do PRE por um dos grupos-piloto. A seção 6.1.1 relata como foram cons-

[8] `http://gcu.dc.ufscar.br/`

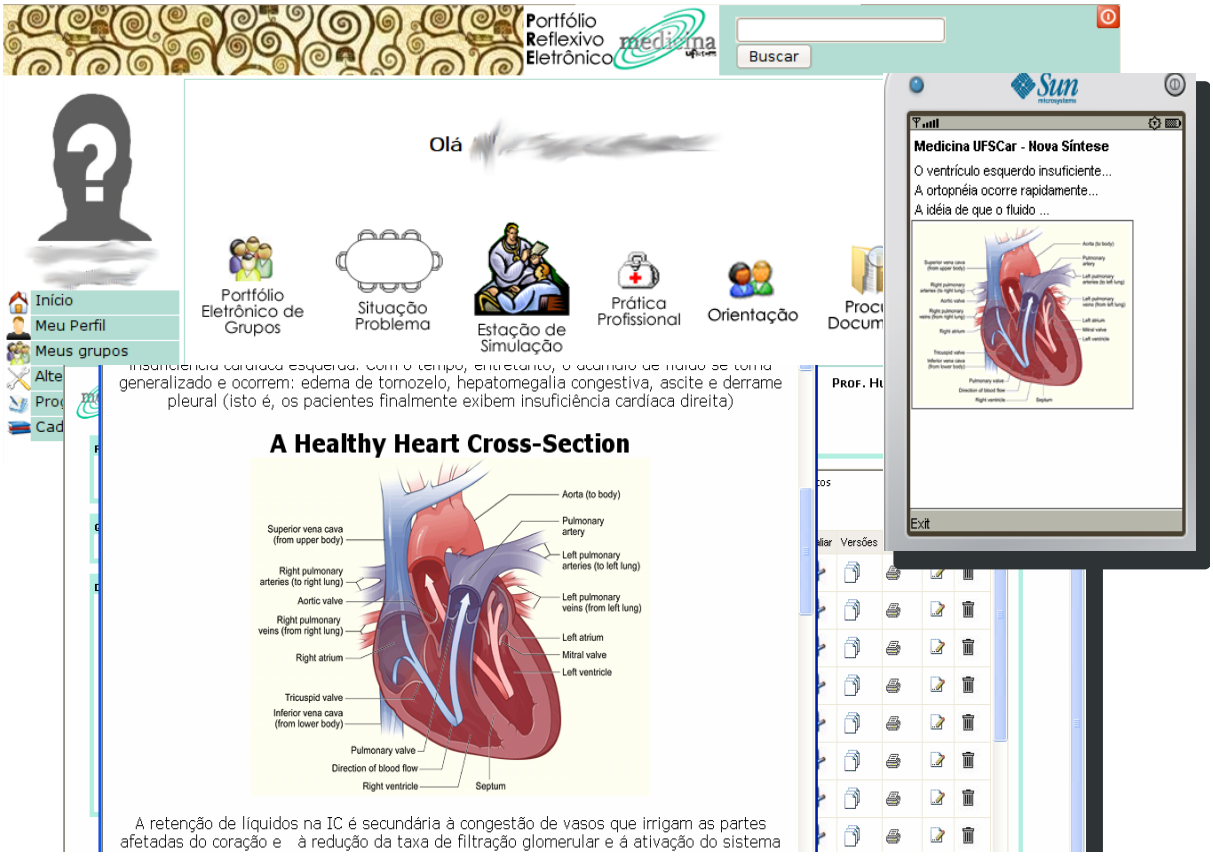

Figura 6.1: Portfolio Reflexivo Eletrônico Ubíquo: versões *desktop* e *mobile*

truídas a coleção de testes e as situações empregadas no estudo de caso; a seção 6.1.2 reporta a execução do experimento e discute os resultados.

6.1.1 Obtenção dos dados

De forma a avaliar este estudo de caso sob uma perspectiva quantitativa, foi necessária a obtenção de um conjunto de dados formado não apenas por uma coleção de documentos, mas também por consultas e julgamentos de relevância, como é comum em experimentos de RI; foi necessária também a obtenção de um conjunto de dados contextuais associados às consultas e aos documentos.

Em face dessa demanda, derivou-se uma pequena coleção de teste baseada nos registros de interação e documentos do ambiente PRE Ubíquo. Os dados considerados foram coletados durante um experimento de avaliação de usabilidade da ferramenta, conduzido ao longo de 10 semanas por um grupo-piloto. Para a obtenção dos documentos foram considerados os documentos coletados pelos usuários ao longo do período. Embora a obtenção dos documentos tenha sido trivial, as consultas, os jul-

gamentos de relevância e os dados contextuais constituiu-se em uma tarefa mais complexa.

Com relação às consultas, foi considerado um gênero especial de documento disponível no sistema, denominado "Questões de Aprendizagem". Documentos desse gênero são formados por um conjunto de questões para as quais o usuário deve prover respostas ao longo das sessões de aprendizado. Uma vez que as respostas são obtidas, usualmente são acompanhadas de referências bibliográficas que as atestem, e essas referências, em muitos casos, são outros documentos armazenados no sistema.

Considerando esse cenário, para o propósito de avaliação experimental, as questões de aprendizagem foram consideradas como consultas e as referências bibliográficas que são documentos do sistema, foram consideradas como julgamentos de relevância. Para obter os dados contextuais que vão compor as situações o ambiente foi configurado para reportar os registros de interação já no formato da ontologia de contexto. As situações foram obtidas em lote pelo algoritmo de manutenção de situações do componente de Coleta da arquitetura.

Após esse processo, realizou-se uma filtragem para eliminar documentos com fraco conteúdo informacional, consultas mal formadas bem como consultas sem julgamentos de relevância usáveis ou nenhuma situação relacionada. Em suma, a coleção de teste constituiu-se dos ítens listados na tabela 6.1.

Tabela 6.1: Estatísticas da coleção de teste PRE

Documentos	687
Consultas	25
Situações	243
Situações de consultas	25
Documentos com situações	379
Média documentos/situação	1.55
Usuários	11

A coleção PRE constitui-se de 687 documentos e 25 consultas. Foi possível obter 243 situações a partir dos registros de interação, sendo 25 delas relacionadas a consultas (uma situação por consulta) e as restantes a documentos, resultando em 379 documentos com ao menos uma situação relacionada. Os registros de interação referem-se às ações de 11 usuários.

6.1.2 Avaliação experimental

O experimento de avaliação consistiu em comparar o desempenho de três configurações distintas da técnica de retroalimentação de relevância com relação a um sistema de referência (*baseline*) sem expansão de consulta. Todos os sistemas empregam o modelo de espaço de vetores. A diferenciação dos sistemas de expansão consistiu na estratégia de seleção de evidências. Foram configurados os seguintes sistemas:

- Sistema NE (*No Evidences*): sistema de referência sem funcionalidades para expansão de consulta
- Sistema PE (*Pseudo Evidences*): as evidências são obtidas do topo da lista de resultados (pseudo retroalimentação de relevância)
- Sistema RE (*Random Evidences*): as evidências são obtidas aleatoriamente do conjunto formado pelos 100 primeiros resultados da consulta original
- Sistema IE (*Implicit Evidences*): as evidências são determinadas implicitamente segundo a abordagem RISC-RIR, usando as situações coletadas do ambiente PRE Ubíquo para selecionar as evidências

Tabela 6.2: Relação de precisão média em 11 níveis de revocação para os sistemas testados: coleção PRE

Revocação	NE	PE	RE	IE
0	0.681	0.701	0.559	0.731
0.1	0.458	0.471	0.316	0.584
0.2	0.389	0.408	0.284	0.493
0.3	0.341	0.372	0.260	0.465
0.4	0.273	0.312	0.222	0.362
0.5	0.182	0.226	0.156	0.306
0.6	0.158	0.213	0.139	0.248
0.7	0.081	0.114	0.074	0.121
0.8	0.037	0.066	0.036	0.073
0.9	0.002	0.006	0.002	0.009
1	0.012	0.013	0.006	0.017
Média	0.238	0.264	0.187	0.310

Todos os sistemas foram configurados utilizando-se o *toolkit* Lemur. No caso dos sistemas PE, RE e IE, foi utilizada a versão do algoritmo de Rochio apresentada na seção 5.2.2, que foi configurado para considerar apenas 10 evidências e expandir a consulta com 3 termos extraídos dessas evidências. As consultas foram emitidas em

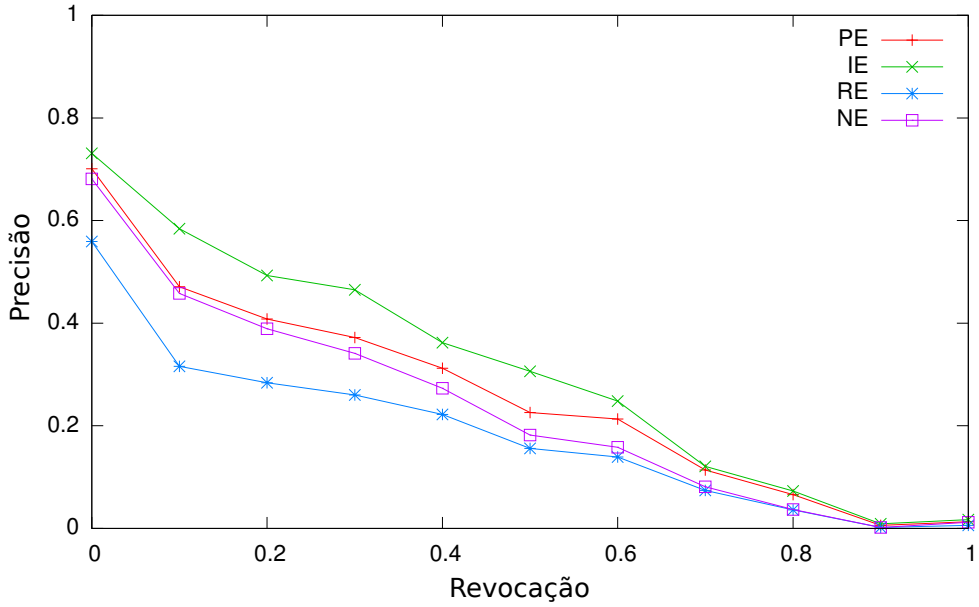

Figura 6.2: Gráfico de precisão média em 11 níveis de revocação para os sistemas testados: coleção PRE

lote nos quatro sistemas e o desempenho de cada um foi mensurado considerando precisão e revocação. Os resultados são apresentados na figura 6.2; os dados que foram plotados são relacionados na tabela 6.2.

É perceptível a partir da figura 6.2 e tabela 6.2 que o desempenho do sistema RE, que emprega evidências aleatórias tomadas do *ranking* de resultados, é expressivamente inferior a todos os restantes, inclusive ao sistema de referência, em todos os níveis de revocação. Tal resultado demonstra que a estratégia de seleção de evidências impacta diretamente na qualidade de expansão de consulta e, consequentemente, no desempenho do sistema, podendo trazer consequências negativas quando não se aplica um processo cuidadoso para conduzir a expansão.

Já o sistema PE, que emprega como evidências os primeiros documentos do *ranking*, apresenta ganho de desempenho marginal nos níveis mais baixos de revocação, e ganhos entre 4% e 6% a partir dos níveis intermediários. Desta constatação derivam-se dois fatos: (a) se o sistema PE apresentou ganhos, as consultas da coleção de teste têm boa qualidade, já que a retroalimentação cega de relevância apresenta bom desempenho com consultas que operam satisfatoriamente; (b) já que o maior ganho de desempenho ocorre a partir dos níveis intermediários de revocação, isso indica que o sistema PE, quando comparado ao NE, retorna mais documentos relevantes no final da lista de resultados, privilegiando os benefícios aos usuários que se dispuserem a inspecionar além dos primeiros resultados.

Analisando a curva do sistema IE, o qual emprega evidências implícitas, nota-se que apresenta desempenho superior a todos os outros sistemas, em todos os níveis de revocação. O ganho é mais notável entre os primeiros níveis de revocação, tornando-se marginal a partir do nível 0.7. Nos primeiros níveis de revocação, a precisão apresenta ganhos entre 5% e 18% em comparação ao sistema de referência. O fato do sistema IE apresentar ganhos em todos os níveis de revocação, por si só, já contribui positivamente para melhor satisfazer as necessidades informacionais dos usuários.

Complementarmente, pela ocorrência de maior precisão nos primeiros níveis de revocação, tem-se que IE retorna mais documentos relevantes no topo do ranking. Esta vantagem favorece especialmente os usuários que não estão dispostos a percorrer longamente a lista de resultados. Considerando usuários de dispositivos móveis, em particular, a obtenção de mais resultados relevantes no topo do *ranking* traz menor necessidade de navegar a paginação dos resultados e menos rolagem na tela do dispositivo, devido ao fato de bons resultados encontrarem-se disponíveis já na primeira tela ou nas telas iniciais. Além disso, a obtenção de mais resultados relevantes diminui as chances do usuário iniciar uma nova iteração para reformular a consulta original, o que reduz a necessidade de digitar novos termos nos teclados restritos dos dispositivos.

Enfim, os resultados demonstram que a abordagem RISC-RIR melhora a qualidade da recuperação de informação, à medida que considera o contexto de trabalho do usuário como um subsídio para melhor interpretar as necessidades informacionais. Adicionalmente, devido ao fato dos ganhos serem mais expressivos nos menores níveis de revocação, obtêm-se mais documentos relevantes no topo do *ranking*. Tal fato conduz à diminuição da sobrecarga cognitiva do usuário em inspecionar os resultados, o que constitui em benefício especialmente útil à interação de usuários de dispositivos móveis com sistemas de RI.

No presente estudo de caso, buscou-se demonstrar e discutir o desempenho da abordagem RISC-RIR com referência a uma pequena coleção de teste derivada de um ambiente de aprendizado de pequeno porte. De forma a averiguar o comportamento da abordagem em condições de maior escala, na próxima seção relata-se o experimento conduzido em um conjunto de dados substancialmente maior, obtidos a partir de um ambiente de aprendizado amplamente usado.

6.2 Estudo de caso: UAB-UFSCar

A Universidade Aberta do Brasil (UAB) é uma iniciativa pública para Educação à Distância (EaD) criada em 2005, articulando universidades federais e governos municipais, com o propósito de interiorizar e democratizar a educação superior no Brasil, levando-a a municípios que ou não possuem ofertas de cursos superiores ou não contam com vagas suficientes para atender às demandas regionais. A UFSCar é uma das instituições federais participantes do projeto UAB, oferecendo cinco cursos de graduação. Cada curso é mantido por um coordenador de curso, que gerencia os professores responsáveis por cada disciplina. O professor, por sua vez, gerencia os tutores virtuais, em média 1 para cada 25 alunos por pólo, que têm contato direto e constante com os alunos durante o processo de aprendizado.

Para a condução das atividades didáticas à distância, a UAB-UFSCar utiliza o ambiente Moodle como CSCL. O ambiente Moodle da UAB-UFSCar disponibiliza a professores, alunos e tutores diversas ferramentas, como comunicador instantâneo, e-mail, fóruns de discussão, wikis, escaninho de tarefas, entre outras. Uma funcionalidade importante do Moodle é o registro de todas as atividades executadas pelos usuários que acessaram as salas virtuais de aula, especificando o tipo de ação executada sobre as ferramentas e objetos informacionais. Esses registros de interação podem ser visualizados por meio de relatórios, que em geral são usados pelos tutores e professores para avaliar o nível de acesso dos alunos em cada sala. Um exemplo de relatório dos registros de interação é apresentado na figura 6.3.

Em vista da riqueza de registros de interação do Moodle, bem como da ampla quantidade de documentos disponíveis num ambiente de EaD de larga escala como o da UAB-UFSCar, dados deste ambiente foram usados em um estudo de caso para avaliar o comportamento da abordagem RISC-RIR frente a demandas maiores de diversidade e número de documentos, bem como maior heterogeneidade de usuários. O restante desta seção está organizado da seguinte forma: a seção 6.2.1 relata o processo de preparação dos dados para o experimento; a seção 6.2.2 expõe a avaliação que foi conduzida e discute os resultados obtidos.

6.2.1 Obtenção dos dados

Como no estudo de caso anterior, a avaliação deste experimento foi conduzida sob uma perspectiva quantitativa, o que demandou a obtenção de uma coleção de teste

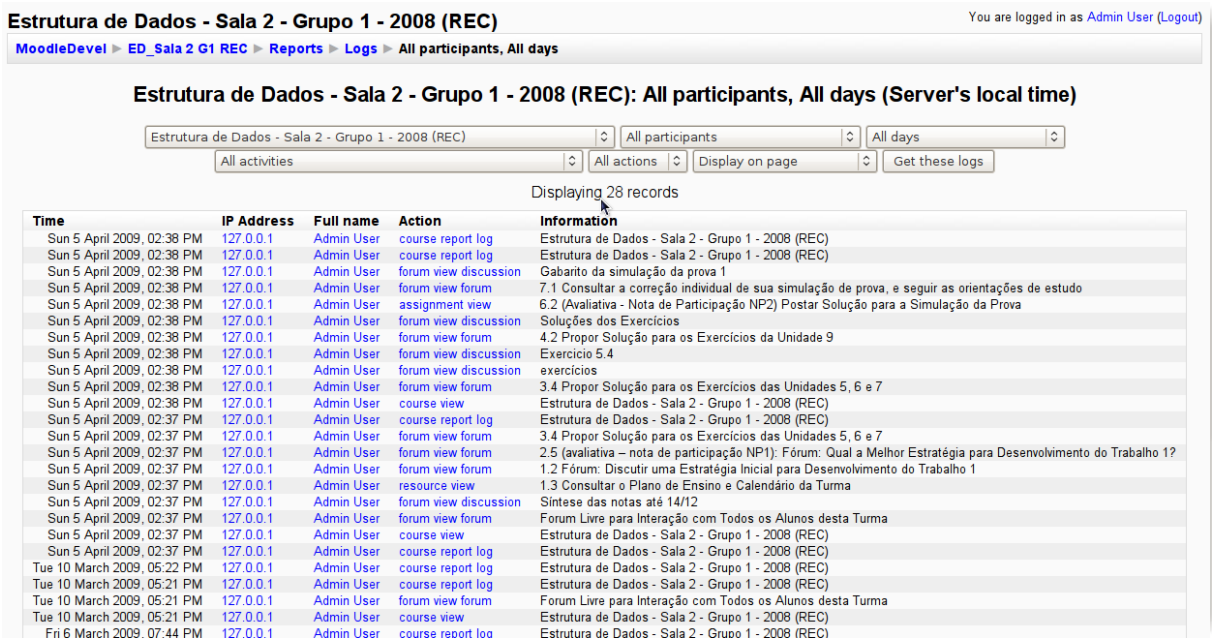

Figura 6.3: Relatório de registros de interação do ambiente Moodle

composta por documentos, consultas, julgamentos de relevância e, adicionalmente, situações associadas às consultas e aos documentos. Para tal, aproveitou-se dos dados obtidos a partir de 22 salas virtuais distribuídas por três cursos da UAB-UFSCar. Cada uma das salas virtuais corresponde a uma disciplina ministrada ao longo de um período letivo (aproximadamente 1 semestre). Foi extraído, a partir de recurso próprio do ambiente Moodle, um pacote de *backup* para cada sala, englobando usuários, documentos e registros de interação.

Os pacotes assim obtidos foram restaurados em instalação local do ambiente Moodle, configurada especialmente para o experimento. A partir dessa instalação, foram necessárias medidas adicionais para obter os documentos, as consultas, os julgamentos de relevância e as situações.

Quanto aos documentos, constatou-se que mantinham-se distribuídos por grande parte das 198 tabelas do banco de dados do ambiente. Deste modo, para conduzir a extração dos documentos adotou-se a estratégia de adaptar o indexador da ferramenta de busca do Moodle para, ao invés de materializar um índice, exportar os documentos e metadados correspondentes para uma localidade pré-determinada do sistema de arquivos.

As consultas, por sua vez, puderam ser obtidas a partir dos registros de interação da ferramenta de busca do Moodle. Do total de 258 consultas, foram filtradas 38 para compor o conjunto final, considerando como critérios a qualidade das consultas (que

deveriam ser curtas e sem erros de digitação), a distribuição uniforme do conjunto ao longo do período de coleta dos dados (para evitar muitas consultas em certos períodos de tempo), e a diversidade de usuários que submeteram as consultas. Após a seleção manual, verificou-se se as consultas retornavam número satisfatório de resultados. Para tal, conduziu-se uma indexação preliminar dos documentos do ambiente; em seguida, as consultas foram emitidas em lote; os resultados retornados permitiram constatar que todas as consultas obtidas retornavam um grande número de resultados.

Para obter os julgamentos de relevância, adotou-se o método de *pooling* (descrito na seção 2.3.1). Para tal, foram configurados três sistemas distintos para indexar a coleção de documentos, todos empregando o modelo de espaço de vetores: Lemur, Lucene[9] e Terrier[10]. Para formar o *pool* de cada consulta, foi considerada a união dos 50 primeiros resultados de cada sistema frente à consulta. Desta forma, após a eliminação de redundâncias, obteve-se uma média de 261 documentos por *pool*. Após análise manual para averiguação da relevância dos itens de cada *pool*, constatou-se que cada consulta obteve em média 32 julgamentos de relevância.

Por fim, a obtenção das situações a partir dos registros de interação do Moodle demandou a especialização do componente de Coleta da arquitetura para atender às especificidades do ambiente. No Moodle, os registros de interação são armazenados no banco de dados do ambiente, logo a conversão dos registros foi realizada de acordo com o algoritmo da figura 5.10, capítulo 5.

Em suma, a coleção de testes final apresentou as características descritas na tabela 6.3.

Tabela 6.3: Estatísticas da coleção de teste UAB

Documentos	19397
Consultas	38
Situações	1462
Situações de consultas	38
Documentos com situações	18508
Média documentos/situação	12.65
Salas virtuais	22
Usuários	747

Foi possível obter 19397 documentos e 38 consultas, distribuídos por 22 salas virtuais. Quanto às situações, obteve-se um total de 1462 grafos nomeados, sendo que 18508

[9] `http://lucene.apache.org/`
[10] `http://ir.dcs.gla.ac.uk/terrier/`

documentos e todas as consultas apresentaram ao menos uma situação associada. Os documentos que não apresentaram situações são itens para os quais os registros de interação não reportavam acessos. Com isso obteve-se uma média de 12.65 documentos associados a cada situação.

6.2.2 Avaliação experimental

Para a avaliação experimental, foi elaborado experimento análogo ao apresentado na seção 6.1.2. Definiu-se um sistema de referência (NE) sem funcionalidades de expansão de consulta e 3 sistemas de comparação (PE, RE e IE) com funcionalidade de expansão de consulta via retroalimentação de relevância, cada um dos três adotando estratégias distintas para seleção de evidências. Vale ressaltar que o sistema IE adota a abordagem desenvolvida no capítulo 5. Os resultados obtidos são listados na tabela 6.4 e figura 6.4.

Tabela 6.4: Listagem dos valores de precisão para os sistemas avaliados com a coleção UAB

Revocação	NE	PE	RE	IE
0	0.514	0.544	0.459	0.565
0.1	0.391	0.410	0.287	0.509
0.2	0.312	0.324	0.269	0.415
0.3	0.267	0.281	0.219	0.401
0.4	0.241	0.245	0.190	0.300
0.5	0.215	0.223	0.170	0.250
0.6	0.169	0.176	0.158	0.200
0.7	0.118	0.125	0.114	0.135
0.8	0.092	0.091	0.089	0.106
0.9	0.055	0.054	0.053	0.062
1	0.027	0.028	0.026	0.039
Média	0.218	0.227	0.185	0.271

Observando o comportamento do sistema RE, nota-se que a seleção aleatória de evidências não é uma boa estratégia de expansão de consulta, mesmo frente a uma coleção de maior porte. O sistema RE manteve desempenho inferior a todos os outros sistemas, em todos os níveis de revocação. Consequentemente, corrobora-se a constatação do experimento anterior, e ressalta-se a importância da adoção de uma estratégia adequada para seleção de evidências para expansão de consulta.

Já quanto ao sistema PE constata-se que apresentou desempenho bastante próximo ao sistema de referência, com ganhos marginais nos níveis iniciais de revocação, entre

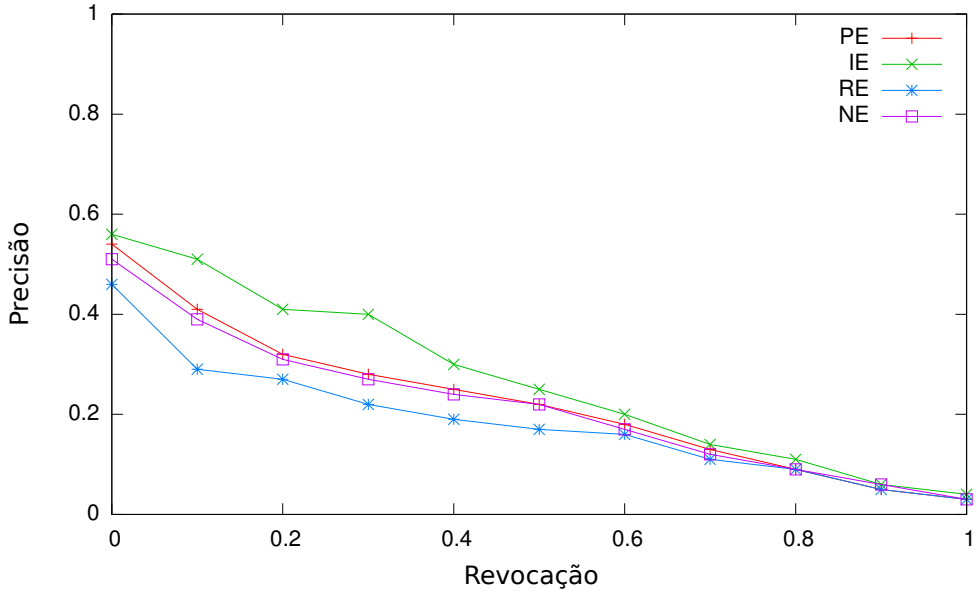

Figura 6.4: Precisão média em 11 níveis de revocação para os sistemas testados: coleção UAB

1% e 4%. Tais ganhos permitem derivar, embora em magnitude mais reduzida do que foi observado no primeiro experimento, que as consultas da coleção e os julgamentos de relevância estão operando satisfatoriamente. Por outro lado, diferentemente do primeiro experimento, os ganhos ocorreram principalmente nos primeiros níveis de revocação, o que desta vez desfavorece a sobrecarga cognitiva do usuário. Mesmo que o sistema PE desfrute de uma estratégia mais simples de seleção de evidências, os ganhos obtidos nos primeiros níveis de revocação são pouco expressivos.

Quando observados os resultados do sistema IE, que adota evidências implícitas, notam-se ganhos de até 14% nos primeiros níveis de revocação, sendo que nos últimos níveis os ganhos em precisão tornam-se marginais. Este comportamento assemelha-se ao observado no experimento anterior. Com comportamento análogo, obtém-se os mesmo benefícios apontados anteriormente, quais sejam, melhor satisfação das necessidades informacionais e menor sobrecarga cognitiva na inspeção de resultados e formulação de consultas.

Em suma, os resultados deste estudo de caso, em particular, demonstram que a abordagem desenvolvida para a expansão de consulta melhora a qualidade dos resultados da recuperação de informação, mesmo com a mudança de variáveis do experimento, tais como a magnitude da coleção de documentos, a diversidade de usuários e situações envolvidas. Conclusões semelhantes também podem ser derivadas quanto à ênfase na obtenção de resultados relevantes no topo da lista de resultados, favorecendo

usuários que interagem com o sistema de RI mediados por dispositivos de capacidades restritas.

Esta seção detalhou e discutiu os resultados do estudo de caso promovido sobre os dados da UAB-UFSCar. As conclusões obtidas permitiram verificar a qualidade do sistema, e nesse sentido obtiveram-se resultados análogos ao do PRE Ubíquo quando comparados aos respectivos sistemas de referência e estratégias alternativas de seleção de evidências para expandir as consultas. Desta forma, o sistema apresenta ganhos à satisfação das necessidades informacionais mesmo sob características adicionais como maior escala da coleção de documentos, heterogeneidade de usuário e situações.

6.3 Considerações finais

Este capítulo apresentou dois estudos de caso com o fim de avaliar o desempenho da abordagem RISC-RIR. O estudo de caso PRE Ubíquo consistiu em um experimento sobre uma coleção de testes de pequeno porte envolvendo situações de um número reduzido de usuários. Já o estudo de caso UAB-UFSCar envolveu um universo expressivamente maior e mais heterogêneo de documentos, usuários e situações. Em ambos os estudos de caso, as abordagens desenvolvidas apresentaram ganhos de desempenho que favorecem a satisfação da necessidades informacionais dos usuários.

7 *Trabalhos correlatos*

Tradicionalmente, as abordagens para retroalimentação implícita de relevância são baseadas na coleta de evidências provenientes da interação direta do usuário com os resultados de busca, isto é, apóiam-se em comportamentos do usuário que permitam derivar o grau de atenção dedicado aos objetos informacionais da lista de resultados. Isso pode ser observado na abordagem de Jung [Jung et al. 2007b], na qual são monitorados os documentos clicados pelo usuário e o tempo de leitura em cada documento; e também nas abordagens de Kelly [Kelly and Teevan 2003] e White [White et al. 2006, White and Kelly 2006, White et al. 2004], em que são considerados operadores de interação adicionais, como seleção, impressão e movimento de olhos. Embora esses trabalhos obtenham resultados promissores, mostram-se reducionistas na coleta de evidências, em dois aspectos: consideram apenas uma fração da tarefa de busca — a inspeção dos resultados; consideram apenas uma dimensão do contexto de busca — a atenção dedicada aos resultados.

Alguns trabalhos buscam ampliar a coleta de evidências para além da inspeção da lista de resultados. Isso pode ser notado na abordagem de Shen [Shen et al. 2005a], que considera, além dos resultados, também análise do histórico de consultas do usuário. Desenvolvimento similar está presente na abordagem de Limbu [Limbu et al. 2006] que, além da análise do histórico de consultas, considera preferências de temas providas explicitamente pelo usuário. Mesmo que busquem ampliar a abrangência da coleta de evidências em relação a diferentes momentos da tarefa de busca, esses trabalhos ainda desprezam dimensões importantes do contexto das necessidades informacionais, a exemplo de eventos observáveis no ambiente de trabalho do usuário.

Uma abordagem mais abrangente que as anteriores é desenvolvida por Teevan [Teevan et al. 2005], que considera os registros de interação com ferramentas da estação de trabalho do usuário, como escrita de email e histórico do navegador. Do ponto de vista do tipo de evidências coletadas, esse trabalho assemelha-se à abordagem RISC-RIR. Porém, o foco de Teevan é a atualização de um modelo de preferências, o que

o torna mais centrado na definição de tópicos preferidos pelo usuário; tal orientação ocorre também nos trabalhos de Harper [Harper and Kelly 2006] e Sieg [Sieg et al. 2007]. A abordagem RISC-RIR, por outro lado, é mais abrangente à medida que busca modelar e armazenar explicitamente o contexto de trabalho dos usuários, ao mesmo tempo que não reproduz os aspectos reducionistas de limitar-se à inspeção e à atenção dedicada à lista de resultados.

Grande parte das abordagens para trabalhar explicitamente com contexto em acesso a informação buscam caracterizar o contexto de trabalho do usuário enquanto interage com o sistema de RI. Tal foco pode ser notado na interface de navegação proposta por Hernandez [Hernandez et al. 2007], com foco em tarefas de astrônomos; e no sistema de recomendação de Redon [Redon et al. 2007], com foco nas tarefas de engenheiros aeronáuticos. Apesar do foco em aproveitamento de representações explícitas de contexto, nenhum desses trabalhos lida diretamente com retroalimentação implícita de relevância para aproveitar esses dados, diferentemente da abordagem RISC-RIR.

A abordagem RISC-RIR provê um mecanismo para retroalimentação implícita de relevância que integra representações explícitas do contexto do usuário num método de raciocínio por analogia para a determinação de evidências. A decisão de usar raciocínio por analogia sobre dados de contexto é subsidiada por trabalhos que usam RBC para processar informações de contexto, a exemplo dos trabalhos de Lee [Lee and Lee 2008] e de Akta [Aktas et al. 2004]. Complementarmente, a premissa de que um documento que mostrou-se útil em uma situação pode ser útil em situações similares é verificada experimentalmente no trabalho de Campbell [Campbell et al. 2007] e foi validada pelos resultados apresentados nesta dissertação.

O próximo capítulo encerra esta dissertação, tratando da discussão final sobre os tópicos abordados no texto e direcionando trabalhos futuros.

8 *Conclusão e trabalhos futuros*

Nesta dissertação apresentou-se o desenvolvimento de uma abordagem para melhorar a qualidade dos resultados de sistemas de recuperação de informação, em especial os que podem ser acessados via dispositivos móveis. A abordagem desenvolvida integra o contexto de trabalho dos usuários em um mecanismo para retroalimentação implícita de relevância, apoiado por uma metodologia de Raciocínio Baseado em Casos, de forma a prover recuperação personalizada de informação. Para alcançar tal objetivo, foi desenvolvida uma arquitetura para gerenciar a transformação e processamento de informações de contexto, e seleção de evidências para expansão de consultas via retroalimentação implícita de relevância.

Como fundamentação teórica, a dissertação apresentou conceitos de Recuperação de Informação, enfatizando seus processos, modelos e metodologias de avaliação. Foi exposta a importância da ciência de contexto para interpretação de necessidades informacionais e determinação de relevância dos documentos recuperados em um sistema de RI, bem como as principais tendências de pesquisa para se trabalhar com essas diretrizes. Tratou-se do problema de modificação de consulta, como solução para minimizar a divergência de vocabulário em sistemas de RI, com ênfase nos métodos de retroalimentação de relevância.

Para avaliar a abordagem desenvolvida, foram conduzidos dois estudos de caso em ambientes de aprendizado eletrônico. O primeiro estudo de caso foi conduzido em um ambiente para aprendizado de medicina baseado em problemas, empregado em grupos-piloto do curso de Medicina da UFSCar. O segundo estudo de caso foi conduzido em dados do ambiente Moodle utilizado pelos cursos de educação à distância da UFSCar (UAB-UFSCar).

Em ambos os estudos de caso, os resultados revelam que a abordagem desenvolvida para retroalimentação implícita de relevância melhora a qualidade da recuperação de informação, à medida que considera o contexto de trabalho do usuário como

um subsídio para melhor interpretar as necessidades informacionais. Os ganhos observados permanecem expressivos mesmo com variações na magnitude da coleção de documentos e diversidade de usuários e situações envolvidas. Adicionalmente, devido ao fato dos ganhos serem mais expressivos nas condições em que há mais documentos relevantes no topo do *ranking* de resultados, diminui-se o esforço do usuário em inspecionar os resultados, o que constitui um benefício útil à interação de usuários de dispositivos de capacidades restritas com sistemas de RI.

Produção científica

Do ponto de vista da produção científica oriunda diretamente desta dissertação, foram obtidos dois trabalhos em eventos internacionais, listados a seguir:

> **Martins, D. S.**, Santana, L. H. Z., Biajiz, M., Prado, A. F., e Souza, W. L. (2008). Context-aware information retrieval on a ubiquitous medical learning environment. In *SAC '08: Proceedings of the 2008 ACM Symposium on Applied Computing*, pages 2348–2349, New York, NY, USA. ACM.

> **Martins, D. S.**, Biajiz, M., do Prado, A. F., e de Souza, W. L. (2009). Implicit relevance feedback for context-aware information retrieval in ubilearning environments. In *SAC '09: Proceedings of the 2009 ACM Symposium on Applied Computing*, pages 659–663, New York, NY, USA. ACM.

O arcabouço geral da pesquisa desta dissertação foi publicado em [Martins et al. 2008]. Este trabalho inclui uma visão geral e de alto nível da arquitetura da abordagem RISC-RIR, bem como o direcionamento de algumas extensões. Tais extensões constituem trabalhos futuros. Os primeiros resultados desta dissertação foram publicados em [Martins et al. 2009], englobando o estudo de caso PRE (Portfólio Reflexivo Eletrônico) Ubíquo.

Do ponto de vista de trabalhos em regime de colaboração, foram publicados os seguintes artigos:

> Perlin, C., Santana, L. H. Z., **Martins, D. S.**, Prado, A. F., Souza, W. L., e Biajiz, M. (2007). Um ambiente de Computação Ubíqüa para o ensino baseado em PBL. In *Anais do XXXIII Conferencia Latinoamericana de Informatica (CLEI 2007)*, pages 1–12, San José, Costa Rica.

Santana, L., **Martins, D. S.**, Forte, M., Souza, W. L., Prado, A. F., Biajiz, M., Knoff, L.. (2007). Serviço de tradução de linguagens de marcação para a Internet. *Anais do XXV Simpósio Brasileiro de Redes de Computadores (SBRC)*, Volume 1, pages 541–554, Belém, Brazil.

Santana, L. H. Z., **Martins, D. S.**, Prado, D. A. F., Souza, D. W. L., e Biajiz, M. (2007). Adaptação de páginas Web para dispositivos móveis. In *XIII Brazilian Symposium on Multimedia and the Web (WebMedia 2007)*, Volume 1, pages 1–8, Gramado, Brazil,

Santos, H. F., Santana, L. H. Z., **Martins, D. S.**, Souza, W. L., Prado, A. F., e Biajiz, M. (2008). A ubiquitous computing environment for medical education. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 1395–1399, New York, USA.

Nos trabalhos relatados acima houve colaboração em aspectos importantes da infra-estrutura do PRE (Portfólio Reflexivo Eletrônico) ubíquo, que foi importante para a motivação e a validação da pesquisa aqui relatada. Em particular, em [Perlin et al. 2007] e em [Santos et al. 2008] são estabelecidas as bases arquiteturais do PRE, bem como avaliações de desempenho e com usuários dos protótipos desenvolvidos. A especificação e validação da infra-estrutura de adaptação de conteúdo, com a finalidade de tornar o PRE acessível por dispositivos móveis, é detalhada em [Santana et al. 2007a] e [Santana et al. 2007b].

Limitações da abordagem e trabalhos futuros

A arquitetura RISC-RIR depende da disponibilidade prévia de registros de interação, pois não provê meios para que diferentes ambientes registrem seus eventos. Desse modo, é necessário que o ambiente em que a abordagem for instanciada já reporte eventos de interação e que seja feita a adaptação da lógica do componente de Coleta para transformar o modelo de dados do ambiente para o modelo da ontologia de contexto. Uma estratégia para tratar esse problema é o desenvolvimento de uma API ou *framework* para tratamento de eventos de interação que possa ser programaticamente integrada no ambiente de destino. Dessa forma, o componente de coleta pode manter-se inalterado pois os eventos poderiam ser reportados diretamente no modelo da ontologia. Isso fará parte de um esforço maior de integração da arquitetura aos

ambientes de destino. Tais medidas facilitariam outro requisito importante, a avaliação da arquitetura com sujeitos humanos, o que viabializaria o estudo de aspectos qualitativos do trabalho desenvolvido.

Outra limitação da abordagem é a dificuldade em efetuar o raciocínio baseado em casos em situações em que há poucas situações armazenadas no repositório. Consequentemente, a qualidade da expansão de consulta é fortemente dependente da quantidade de dados de contexto disponível. Tal problema é frequentemente discutido na literatura de sistemas de recomendação, na qual é denominado de problema de partida fria (*cold start problem*). Um meio de driblar essa limitação é o estabelecimento de uma estratégia híbrida que combine evidências explícitas e implícitas, de tal modo que as evidências providas explicitamente pelo usuário possam em certo grau aliviar a ausência de dados de contexto.

Conforme relatado em [Martins et al. 2008], o projeto da abordagem RISC-RIR originalmente prevê o desenvolvimento de algumas extensões, dentre os quais apontam-se: serviços de extração de informação, apoiados por ontologias terminológicas, para prover indexação semântica dos documentos; apresentação adaptativa dos resultados de busca (em taxonomias, em nuvens de conceitos, etc.), com atenção aos recursos dos dispositivos de acesso; estratégias híbridas de retroalimentação de relevância, que enriqueçam o processo implícito com a possibilidade do usuário prover retroalimentação explícita (por exemplo via exploração incremental da lista adaptativa de resultados).

É importante ressaltar que, em abrangência, a ciência de contexto oferece diversas possibilidades para prover acesso personalizado a informação. Nesta dissertação, buscou-se tratar em profundidade o problema de personalização do acesso a objetos informacionais de natureza textual via expansão automática de consulta. Na direção de complementar esses resultados, a pesquisa está atualmente sendo estendida em nível de doutorado, com a exploração de aspectos que não foram cobertos nesta dissertação, tais como: direcionamentos em interação usuário-computador; recuperação de informações multimídia, ao invés de somente texto; modelos de contexto para englobar informações de ambientes instrumentados, com conjuntos de dados de contexto mais abrangentes; e investigação de outras modalidades para acesso à informação, a exemplo de sistemas de recomendação.

Referências Bibliográficas

[Aamodt and Plaza 1994] Aamodt, A. e Plaza, E. (1994). Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Commun.*, 7(1):39–59.

[Abowd and Mynatt 2000] Abowd, G. e Mynatt, E. (2000). Charting Past, Present, and Future Research in Ubiquitous Computing. *ACM Transactions on Computer-Human Interaction*, 7(1):29–58.

[Adomavicius et al. 2005] Adomavicius, G., Sankaranarayanan, R., et al. (2005). Incorporating Contextual Information in Recommender Systems using a Multidimensional Approach. *ACM Transactions on Information Systems*, 23(1):103–145.

[Adomavicius and Tuzhilin 2005] Adomavicius, G. e Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749.

[Agichtein et al. 2007] Agichtein, E., Burges, C., e Brill, E. (2007). Question answering over implicitly structured web content. In *WI '07: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, pages 18–25, Washington, DC, USA. IEEE Computer Society.

[Aktas et al. 2004] Aktas, M. S., Pierce, M., Fox, G. C., e Leake, D. (2004). A web based conversational case-based recommender system for ontology aided metadata discovery. In *GRID '04: Proceedings of the Fifth IEEE/ACM International Workshop on Grid Computing (GRID'04)*, pages 69–75, Washington, DC, USA. IEEE Computer Society.

[Allan et al. 2003] Allan, J., Aslam, J., Belkin, N., e Collaborators (2003). Challenges in information retrieval and language modeling. *SIGIR Forum*, 37(1):31–47.

[Auer et al. 2009] Auer, S., Dietzold, S., Lehmann, J., Hellmann, S., e Aumueller, D. (2009). Triplify: light-weight linked data publication from relational databases. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 621–630, New York, NY, USA. ACM.

[Baeza-Yates and Ribeiro-Neto 1999] Baeza-Yates, R. e Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison Wesley.

[Bai et al. 2008] Bai, Y., Yang, J., e Qiu, Y. (2008). Ontocbr: Ontology-based cbr in context-aware applications. *Multimedia and Ubiquitous Engineering, International Conference on*, 0:164–169.

[Barbosa et al. 2007] Barbosa, J., Hahn, R., S., R., e Barbosa, D. N. F. (2007). Distribuição de conteúdo em ambientes conscientes de contexto. In *13th Brazilian Symposium on Multimedia and the Web*, pages 73–80.

[Bazire and Brézillon 2005] Bazire, M. e Brézillon, P. (2005). *Understanding Context Before Using It*.

[Bhogal et al. 2007] Bhogal, J., Macfarlane, A., e Smith, P. (2007). A review of ontology based query expansion. *Information Processing & Management*, 43(4):866–886.

[Billerbeck and Zobel 2004] Billerbeck, B. e Zobel, J. (2004). Techniques for Efficient Query Expansion. *String Processing and Information Retrieval: 11th International Conference, SPIRE 2004, Padova, Italy, October 5-8, 2004: Proceedings*.

[Bolchini et al. 2007] Bolchini, C., Curino, C. A., Quintarelli, E., Schreiber, F. A., e Tanca, L. (2007). A data-oriented survey of context models. *SIGMOD Rec.*, 36(4):19–26.

[Borlund 2003a] Borlund, P. (2003a). The concept of relevance in ir. *Journal of the American Society for Information Science and Technology*, 54(10):913–925.

[Borlund 2003b] Borlund, P. (2003b). The IIR evaluation model: a framework for evaluation of interactive information retrieval systems. *Information Research*, 8(3):8–3.

[Bouquet et al. 2005] Bouquet, P., Serafini, L., e Stoermer, H. (2005). Introducing Context into RDF Knowledge Bases. In *Proceedings of SWAP 2005, the 2nd Italian Semantic Web Workshop, Trento, Italy, December 14-16, 2005. CEUR Workshop Proceedings, ISSN 1613-0073, online http://ceur-ws.org/Vol-166/70.pdf*.

[Bouzeghoub et al. 2007] Bouzeghoub, A., Do, K., e Lecocq, C. (2007). A situation-based delivery of learning resources in pervasive learning. pages 450–456.

[Brusilovsky 2003] Brusilovsky, P. (2003). Developing adaptive educational hypermedia systems: From design models to authoring tools. *Authoring Tools for Advanced Technology Learning Environment. Dordrecht: Kluwer Academic Publishers*, pages 377–409.

[Brusilovsky et al. 2007] Brusilovsky, P., Kobsa, A., e Nejdl, W., editors (2007). *The Adaptive Web: Methods and Strategies of Web Personalization*, volume 4321 of *Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg.

[Büttcher et al. 2006] Büttcher, S., Clarke, C., e Lushman, B. (2006). Term proximity scoring for ad-hoc retrieval on very large text collections. *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 621–622.

[Byström and Hansen 2005] Byström, K. e Hansen, P. (2005). Conceptual framework for tasks in information studies. *Journal of the American Society for Information Science and Technology*, 56(10):1050–1061.

[Campbell et al. 2007] Campbell, D. R., Culley, S. J., Mcmahon, C. A., e Sellini, F. (2007). An approach for the capture of context-dependent document relationships extracted from bayesian analysis of users' interactions with information. *Information Retrieval*, 10(2):115–141.

[Carroll et al. 2005] Carroll, J. J., Bizer, C., Hayes, P., e Stickler, P. (2005). Named graphs, provenance and trust. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 613–622, New York, NY, USA. ACM Press.

[Chakrabarti 2002] Chakrabarti, S. (2002). *Mining the Web: Analysis of Hypertext and Semi Structured Data*. Morgan Kaufmann.

[Chen et al. 2003] Chen, H., Finin, T., e Joshi, A. (2003). An ontology for context-aware pervasive computing environments.

[Chowdhury 2003] Chowdhury, G. G. (2003). *Introduction to Modern Information Retrieval*. Neal-Schuman Publishers.

[Cosijn and Ingwersen 2000] Cosijn, E. e Ingwersen, P. (2000). Dimensions of relevance. *Information Processing & Management*, 36(4):533–550.

[Crestani and Ruthven 2007] Crestani, F. e Ruthven, I. (2007). Introduction to special issue on contextual information retrieval systems. *Information Retrieval*, 10(2):111–113.

[Dey 2001] Dey, A. (2001). Understanding and Using Context. *Personal and Ubiquitous Computing*, 5(1):4–7.

[Dey and Abowd 1999] Dey, A. e Abowd, G. (1999). Towards a Better Understanding of Context and Context-Awareness. Technical report, GVU Technical Report GIT-GVU-99-22. College of Computing, Georgia Institute of Technology,(1999).

[Ehrig et al. 2005] Ehrig, M., Haase, P., Stojanovic, N., e Hefke, M. (2005). Similarity for ontologies-a comprehensive framework. *13th European Conference on Information Systems*.

[Fischer and Ye 2001] Fischer, G. e Ye, Y. (2001). Exploiting context to make delivered information relevant to tasks and users. *Workshop on User Modeling for Context-Aware Applications, 8th International Conference on User Modeling (UM2001)*.

[Foltz 1996] Foltz, P. (1996). Latent semantic analysis for text-based research. *Behavior Research Methods, Instruments and Computers*, 28(2):197–202.

[Freund and Toms 2005] Freund, L. e Toms, E. G. (2005). Contextual search: from information behaviour to information retrieval. *Proceedings of the Annual Conference of the Canadian Association for Information Science*.

[Freund et al. 2005] Freund, L., Toms, E. G., e Clarke, C. L. A. (2005). Modeling task-genre relationships for ir in the workplace. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 441–448, New York, NY, USA. ACM.

[Gauch et al. 2003] Gauch, S., Chaffee, J., e Pretschner, A. (2003). Ontology-based personalized search and browsing. *Web Intelligence and Agent Systems*, 1(3-4):219–234.

[Goker and Myrhaug 2008] Goker, A. e Myrhaug, H. (2008). Evaluation of a mobile information system in context. *Information Processing & Management*, 44(1):39–65.

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

[Guarino 1998] Guarino, N. (1998). *Formal ontology in information systems*. IOS Press.

[Hanani et al. 2001] Hanani, U., Shapira, B., e Shoval, P. (2001). Information filtering: Overview of issues, research and systems. *User Modeling and User-Adapted Interaction*, 11(3):203–259.

[Harper and Kelly 2006] Harper, D. J. e Kelly, D. (2006). Contextual relevance feedback. In *IIiX: Proceedings of the 1st international conference on Information interaction in context*, pages 129–137, New York, NY, USA. ACM Press.

[Henricksen et al. 2005] Henricksen, K., Indulska, J., Mcfadden, T., e Balasubramaniam, S. (2005). Middleware for distributed context-aware systems.

[Hernandez et al. 2007] Hernandez, N., Mothe, J., Chrisment, C., e Egret, D. (2007). Modeling context through domain ontologies. *Information Retrieval*, V10(2):143–172.

[Hu and Moore 2007] Hu, B. e Moore, P. (2007). "smartcontext": An ontology based context model for cooperative mobile learning. pages 717–726.

[Ingwersen 1992a] Ingwersen, P. (1992a). *Information Retrieval Interaction*. Taylor Graham, London.

[Ingwersen 1992b] Ingwersen, P. (1992b). *Information Retrieval Interaction*. Taylor Graham, London.

[Ingwersen and Järvelin 2005] Ingwersen, P. e Järvelin, K. (2005). *The Turn: Integration of Information Seeking and Retrieval in Context*. Kluwer Academic Pub.

[Jansen, Bernard J. and Pooch, Udo 2001] Jansen, Bernard J. and Pooch, Udo (2001). A review of web searching studies and a framework for future research. *Journal of the Am. Soc. on Inf. Science and Tech.*, 52(3):235–246.

[Järvelin 2007] Järvelin, K. (2007). An analysis of two approaches in information retrieval: From frameworks to study designs. *Journal of the American Society for Information Science and Technology*, 58(7):971–986.

[Järvelin and Ingwersen 2004] Järvelin, K. e Ingwersen, P. (2004). Information seeking research needs extension towards tasks and technology. *Information Research*, 10(1):10–1.

[Joachims 2002] Joachims, T. (2002). Optimizing search engines using clickthrough data. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142, New York, NY, USA. ACM.

[Jung et al. 2007a] Jung, S., Herlocker, J. L., e Webster, J. (2007a). Click data as implicit relevance feedback in web search. *Information Processing and Management*, 43(3):791–807.

[Jung et al. 2007b] Jung, S., Herlocker, J. L., e Webster, J. (2007b). Click data as implicit relevance feedback in web search. *Inf. Process. Manage.*, 43(3):791–807.

[Kazai et al. 2003] Kazai, G., Govert, N., Lalmas, M., e Fuhr, N. (2003). The INEX evaluation initiative. *Intelligent Search on XML Data*, 2818:279–293.

[Kelly and Teevan 2003] Kelly, D. e Teevan, J. (2003). Implicit feedback for inferring user preference: a bibliography. *ACM SIGIR Forum*, 37(2):18–28.

[Kiefer et al. 2008] Kiefer, C., Bernstein, A., e Stocker, M. (2008). The fundamentals of isparql: A virtual triple approach for similarity-based semantic web tasks. pages 295–309.

[Kolodner 1993] Kolodner, J. (1993). *Case-based reasoning*. Morgan Kaufmann Publishers Inc.

[Koshman 2006] Koshman, S. (2006). Visualization-based information retrieval on the web. *Library & Information Science Research*, 28(2):192–207.

[Kraaij 2005] Kraaij, W. (2005). Variations on language modeling for information retrieval. *ACM SIGIR Forum*, 39(1):61–61.

[Lassila et al. 1999] Lassila, O., Swick, R., et al. (1999). Resource description framework (rdf) model and syntax specification.

[Lee and Lee 2008] Lee, J. e Lee, J. (2008). Context awareness by case-based reasoning in a music recommendation system. pages 45–58.

[Lee et al. 2001] Lee, T., Hendler, J., Lassila, O., et al. (2001). The semantic web. *Scientific American*, 284(5):34–43.

[Li and Belkin 2008] Li, Y. e Belkin, N. J. (2008). A faceted approach to conceptualizing tasks in information seeking. *Information Processing & Management*, In Press, Corrected Proof.

[Limbu et al. 2006] Limbu, D. K., Connor, A., et al. (2006). Contextual Relevance Feedback in Web Information Retrieval. In *Proc. of the 1st Int. Conference on Information Interaction in Context (IIiX)*, pages 138–143. ACM Press.

[Lin 2007] Lin, J. (2007). User simulations for evaluating answers to question series. *Information Processing & Management*, 43(3):717–729.

[Maedche and Staab 2002] Maedche, A. e Staab, S. (2002). *Measuring Similarity between Ontologies*.

[Manning et al. 2008] Manning, C. D., Raghavan, P., e Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

[Martins et al. 2009] Martins, D. S., Biajiz, M., do Prado, A. F., e de Souza, W. L. (2009). Implicit relevance feedback for context-aware information retrieval in ubilearning environments. In *SAC '09: Proceedings of the 2009 ACM symposium on Applied Computing*, pages 659–663, New York, NY, USA. ACM.

[Martins et al. 2008] Martins, D. S., Santana, L. H. Z., Biajiz, M., Prado, A. F., e Souza, W. L. (2008). Context-aware information retrieval on a ubiquitous medical learning environment. In *SAC '08: Proc. of the 2008 ACM Symp. on Applied Computing*, pages 2348–2349, New York, NY, USA. ACM.

[Mcdonald and Chen 2006] Mcdonald, D. M. e Chen, H. (2006). Summary in context: Searching versus browsing. *ACM Trans. Inf. Syst.*, 24(1):111–141.

[McGuinness et al. 2004] McGuinness, D., Van Harmelen, F., et al. (2004). Owl web ontology language overview. *W3C recommendation*, 10:2004–03.

[Mechmache et al.] Mechmache, B. F. Z., Boughanem, M., e Alimazighi, Z. Possibility and necessity measures for relevance assessment. In *PIKM '07: Proceedings of the ACM first Ph.D. workshop in CIKM*, pages 155–162, New York, NY, USA. ACM.

[Mitchel 1997] Mitchel, T. (1997). Machine learning. *Machine Learning*, 48(1).

[Mostefaoui et al. 2004] Mostefaoui, G., Pasquier-Rocha, J., e Brezillon, P. (2004). Context-Aware Computing: A Guide for the Pervasive Computing Community. *Pervasive Services, 2004. ICPS 2004. IEEE/ACS International Conference on*, pages 39–48.

[Muresan 2006] Muresan, G. (2006). An investigation of query expansion terms. *Proceedings of the American Society for Information Science and Technology*, 43(1):1–5.

[Mylonas et al. 2008] Mylonas, P., Vallet, D., Castells, P., FernÁndez, M., e Avrithis, Y. (2008). Personalized information retrieval based on context and ontological knowledge. *Knowl. Eng. Rev.*, 23(1):73–100.

[Ogata and Yano 2004] Ogata, H. e Yano, Y. (2004). Context-Aware Support for Computer-Supported Ubiquitous Learning. In *Proc. of the 2nd IEEE Int. Work. on Wireless and Mobile Tech. in Education (WMTE'04)*, page 27. IEEE Computer Society.

[Pal and Shiu 2004] Pal, S. e Shiu, S. (2004). *Foundations of soft case-based reasoning*. Wiley-Interscience.

[Pérez et al. 2006] Pérez, J., Arenas, M., e Gutierrez, C. (2006). Semantics and complexity of sparql. pages 30–43.

[Perlin et al. 2007] Perlin, C., Santana, L. H. Z., Martins, D. S., Prado, A. F., Souza, W. L., e Biajiz, M. (2007). Um ambiente de computação ubíqüa para o ensino baseado em pbl. In *Anais do XXXIII Conferencia Latinoamericana de Informatica (CLEI 2007)*, pages 1–12, San José, Costa Rica.

[Petersen 2006] Petersen, A. K. (2006). Challenges in case-based reasoning for context awareness in ambient intelligent systems. In Minor, M., editor, *8th European Conference on Case-Based Reasoning, Workshop Proceedings*, pages 287–299, Ölüdeniz/-Fethiye, Turkey.

[Petersen and Cassens 2006] Petersen, A. K. e Cassens, J. (2006). Using activity theory to model context awareness. In *Modeling and Retrieval of Context*, volume 3946, pages 1–17. Springer.

[Recio-Garcıa et al. 2006] Recio-Garcıa, J., Dıaz-Agudo, B., Gonzalez-Calero, P., e Sanchez, A. (2006). Ontology based CBR with jCOLIBRI. *Applications and Innovations in Intelligent Systems*, 14:149–162.

[Redon et al. 2007] Redon, R., Larsson, A., Leblond, R., e Longueville, B. (2007). Vivace context based search platform. *Modeling and Using Context*, pages 397–410.

[Rigo and Oliveira 2007] Rigo, S. J. e Oliveira, J. P. M. d. (2007). Personalização de sítios web integrando mineração de uso e ontologias. In *13th Brazilian Symposium on Multimedia and the Web*, pages 151–158.

[Rocchio 1971] Rocchio, J. (1971). Relevance feedback in information retrieval. *The SMART retrieval system: experiments in automatic document processing*, pages 313–323.

[Ruthven and Lalmas 2003] Ruthven, I. e Lalmas, M. (2003). A Survey on the Use of Relevance Feedback for Information Access Systems. *Knowledge Engeneering Review*, 18(2):95–145.

[Salton and Buckley 1988] Salton, G. e Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523.

[Salton and Buckley 1990] Salton, G. e Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297.

[Salton et al. 1975] Salton, G., Wong, A., e Yang, C. S. (1975). A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620.

[Santana et al. 2007a] Santana, L., Martins, D., Forte, M., et al. (2007a). Serviço de tradução de linguagens de marcação para a Internet. *Anais do XXV Simpósio Brasileiro de Redes de Computadores*, 1:541–554.

[Santana et al. 2007b] Santana, L. H. Z., Martins, D. S., et al. (2007b). Adaptação de Páginas Web para Dispositivos Móveis. In *Proc. of the 13th Brazilian Symposium on Multimedia and the Web (Webmedia '07)*. ACM Press.

[Santana et al. 2007c] Santana, L. H. Z., Martins, D. S., Prado, D. A. F., Souza, D. W. L., e Biajiz, M. (2007c). Adaptação de páginas web para dispositivos móveis. In Sb, C., editor, *XIII Brazilian Symposium on Multimedia and the Web (WebMedia 2007)*, volume 1, pages 1–8.

[Santos et al. 2008] Santos, H. F., Santana, L. H. Z., Martins, D. S., Souza, W. L., Prado, A. F., e Biajiz, M. (2008). A ubiquitous computing environment for medical education. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 1395–1399, New York, NY, USA. ACM.

[Shadbolt et al. 2006] Shadbolt, N., Hall, W., e Berners-Lee, T. (2006). The Semantic Web Revisited. *IEEE Intelligent Systems*, 21(3):96–101.

[Shen et al. 2005a] Shen, X., Tan, B., e Zhai, C. (2005a). Context-sensitive information retrieval using implicit feedback. In *SIGIR '05: Proc. of the 28th ACM SIGIR Conf. on Research and Development in Inf. Retrieval*, pages 43–50. ACM Press.

[Shen et al. 2005b] Shen, X., Tan, B., e Zhai, C. (2005b). Context-sensitive information retrieval using implicit feedback. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 43–50, New York, NY, USA. ACM Press.

[Sieg et al. 2007] Sieg, A., Mobasher, B., e Burke, R. (2007). Web search personalization with ontological user profiles. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 525–534, New York, NY, USA. ACM.

[Silva and Favela 2006] Silva, J. M. e Favela, J. (2006). Context aware retrieval of health information on the web. In *LA-WEB '06: Proceedings of the Fourth Latin American Web Congress*, pages 135–146, Washington, DC, USA. IEEE Computer Society.

[Stajano 2002] Stajano, F. (2002). *Security for Ubiquitous Computing*. John Wiley & Sons.

[Stoermer et al. 2006] Stoermer, H., Palmisano, I., Redavid, D., Iannone, L., Bouquet, P., e Semeraro, G. (2006). RDF and Contexts: Use of SPARQL and Named Graphs to Achieve Contextualization. *Proccedings of the 2006 Jena User Conference*.

[Strang and Linnhoff-Popien 2004] Strang, T. e Linnhoff-Popien, C. (2004). A Context Modeling Survey. In *Workshop on Advanced Context Modelling, Reasoning and Management. Proc. of the 6th Int. Conference on Ubiquitous Computing (UbiComp 2004)*.

[Teevan et al. 2005] Teevan, J., Dumais, S. T., e Horvitz, E. (2005). Personalizing search via automated analysis of interests and activities. In *SIGIR '05: Proceedings of the 28th ACM SIGIR Conf. on Research and Development in Inf. Retrieval*, pages 449–456. ACM Press.

[Traina et al. 2006] Traina, A. J. M., Marques, J., e Traina, C. (2006). Fighting the semantic gap on cbir systems through new relevance feedback techniques. *Computer-Based Medical Systems, IEEE Symposium on*, 0:881–886.

[Vinay et al. 2005] Vinay, V., Wood, K., Milic-Frayling, N., e Cox, I. J. (2005). Comparing relevance feedback algorithms for web search. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 1052–1053, New York, NY, USA. ACM.

[Voorhees 2005] Voorhees, E. M. (2005). *TREC: Experiment and Evaluation in Information Retrieval*. Digital Libraries and Electronic Publishing. MIT Press.

[Wang et al. 2006] Wang, G., Jiang, J., e Shi, M. (2006). A context model for collaborative environment. pages 1–6.

[Weiser 1999a] Weiser, M. (1999a). Some computer science issues in ubiquitous computing. *ACM SIGMOBILE Mobile Computing and Communications Review*, 3(3).

[Weiser 1999b] Weiser, M. (1999b). The computer for the 21 stcentury. *ACM SIGMOBILE Mobile Computing and Communications Review*, 3(3):3–11.

[White et al. 2004] White, R., Jose, J., van Rijsbergen, C., e Ruthven, I. (2004). A simulated study of implicit feedback models.

[White et al. 2006] White, R. W., Jose, J. M., e Ruthven, I. (2006). An implicit feedback approach for interactive information retrieval. *Inf. Process. Manage.*, 42(1):166–190.

[White and Kelly 2006] White, R. W. e Kelly, D. (2006). A study on the effects of personalization and task information on implicit feedback performance. In *CIKM '06: Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 297–306, New York, NY, USA. ACM Press.

[Whittle et al. 2007] Whittle, M., Eaglestone, B., Ford, N., Gillet, V., e Madden, A. (2007). Data mining of search engine logs. *Journal of the American Society for Information Science and Technology*, 58(14):2382–2400.

[Wilkinson and Wu 2004] Wilkinson, R. e Wu, M. (2004). Evaluation Experiments and Experience from the Perspective of Interactive Information Retrieval. *the Proceedings of the Third Workshop on Empirical Evaluation of Adaptive Systems, in conjunction with AH2004. August*, pages 23–26.

[Wong et al. 2008] Wong, W. S., Luk, R. W. P., Leong, H. V., Ho, K. S., e Lee, D. L. (2008). Re-examining the effects of adding relevance information in a relevance feedback environment. *Information Processing & Management*, In Press, Corrected Proof.

[Xu and Croft 1996] Xu, J. e Croft, B. W. (1996). Query expansion using local and global document analysis. In *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11.

[Xu and Chen 2006] Xu, Y. e Chen, Z. (2006). Relevance judgment: What do information users consider beyond topicality? *Journal of the American Society for Information Science and Technology*, 57(7):961–973.

[Yang et al. 2006] Yang, S. J. H., Huang, A. F. M., Chen, R., Tseng, S. S., e Shen, Y. S. (2006). Context model and context acquisition for ubiquitous content access in ulearning environments. In *SUTC '06: Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing - Vol 2 - Workshops*, pages 78–83, Washington, DC, USA. IEEE Computer Society.

[Zimmermann 2003] Zimmermann, A. (2003). Context-awareness in user modelling: Requirements analysis for a case-based reasoning application. page 1064.