

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

GUILHERME MEDEIROS MACHADO

**Recomendação Adaptativa e Sensível ao
Contexto de Recursos para Usuários em um
Campus Universitário**

Dissertação apresentada como requisito parcial
para a obtenção do grau de Mestre em Ciência
da Computação

Prof. Dr. José Palazzo Moreira de Oliveira
Orientador

Porto Alegre, fevereiro de 2014.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Machado, Guilherme Medeiros

Recomendação Adaptativa e Sensível ao Contexto de Recursos para Usuários Móveis em um Campus Universitário [manuscrito] / Guilherme Medeiros Machado. – 2014.

67 f.:il.

Orientador: José Palazzo Moreira de Oliveira.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2014.

1.Recomendação de Recursos. 2.Ambientes Pervasivos. 3.Sensibilidade ao Contexto. I. Palazzo Moreira de Oliveira, José. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitor de Pós-Graduação: Prof. Vladimir Pinheiro do Nascimento

Diretor do Instituto de Informática: Prof. Luís da Cunha Lamb

Coordenador do PPGC: Prof. Luigi Carro

Bibliotecário-Chefe do Instituto de Informática: Alexander Borges Ribeiro

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter me fortalecido durante toda elaboração deste trabalho. Agradeço também a toda a minha família, especialmente minha mãe, pai e irmã por terem estado sempre ao meu lado me apoiando durante esses dois anos de mestrado e agora rumo ao doutorado.

Agradeço especialmente ao meu orientador Prof. Dr. José Palazzo, por estar sempre disponível, pelo profissionalismo, pelo incentivo, pela paciência, por todas orientações das quais este trabalho dependeu, um exemplo que pretendo seguir em minha carreira acadêmica.

Agradeço aos amigos que estiveram presentes tanto em momentos de calma quanto nos de turbulência durante o desenvolvimento desta dissertação, em especial: Júlia, Diego, Wagner e Leo, Jonas, Ana e Isabela, amigos que me ajudaram em assuntos acadêmicos e nunca me deixaram só, me ajudando a suportar a saudade e a distância da família. Além de todos os outros colegas do laboratório 213, que me acompanharam nestes anos.

Agradeço à UFRGS e todos os professores e técnicos administrativos que me auxiliaram durante as aulas ou durante problemas administrativos. Agradeço também ao CNPq e a CAPES por financiarem parcialmente esta pesquisa.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS.....	5
LISTA DE FIGURAS	6
LISTA DE TABELAS	7
RESUMO.....	8
ABSTRACT.....	9
1 INTRODUÇÃO	10
2 MODELAGEM DE CONTEXTO	14
2.1 Técnicas de Modelagem de Contexto.....	15
2.1.1 Baseadas em papéis objeto	16
2.1.2 Baseadas em modelos espaciais	16
2.1.3 Baseadas em ontologias	16
2.2 Trabalhos Relacionados	17
2.2.1 Modelos de Alto Nível	17
2.2.2 Modelos aplicados	20
2.3 Considerações Finais	21
3 SISTEMAS DE RECOMENDAÇÃO E SENSIBILIDADE AO CONTEXTO	22
3.1 Recomendação Baseada em Filtragem Colaborativa.....	24
3.2 Recomendação Sensível ao Contexto	25
3.2.1 Agregação de informações	27
3.2.2 Incorporando contexto aos sistemas de recomendação	28
3.3 Trabalhos Relacionados	29
3.4 Considerações Finais	31
4 ABORDAGEM SENSÍVEL AO CONTEXTO PARA RECOMENDAÇÃO DE RECURSOS EM AMBIENTES PERVASIVOS.....	32
4.1 Especificação da Solução	32
4.1.1 Funcionalidades do sistema	33
4.2 Interface da Solução	34
4.3 Modelo Desenvolvido.....	37
4.3.1 Metodologia utilizada	38
4.3.2 Estrutura do modelo	40
4.3.3 Modelos reutilizados	46
4.4 Arquitetura da Solução.....	50
4.5 Considerações Finais	52
5 ESTUDO DE CASO	54
5.1 Cenário de Uso	54
5.2 Motores de Inferência	56
5.3 Regras de Recomendação	60
5.4 Considerações Finais	63
6 CONCLUSÃO.....	64
REFERÊNCIAS.....	66

LISTA DE ABREVIATURAS E SIGLAS

IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
LOM	Learning Objects Metadata
LUBM	Lehigh University Benchmark
OA	Objeto de Aprendizagem
OWL	Web Ontology Language
RDF	Resource Description Framework

LISTA DE FIGURAS

Figura 1.1: Pessoas e variação do número de pessoas de 10 anos ou mais de idade que tinham telefone móvel celular para uso pessoal - Brasil - 2005/2011.....	11
Figura 2.1: Definição parcial da ontologia superior CONON. Adaptado de Wang et al. (2004)	18
Figura 2.2: Ontologia de Pantsar-Syvaniemi distribuída nos níveis de contexto. Adaptado de Pantsar-Syvaniemi, Simula e Ovaska (2010).....	19
Figura 3.1: O processo de filtragem colaborativa. Adaptado de Sarwar et al., (2001) ..	25
Figura 3.2: Espaço de recomendação do modelo multidimensional para usuário x item x tempo. Adaptado de Adomavicius et al., (2005)	26
Figura 3.3: Paradigmas para incorporar informação contextual em sistemas que utilizam a abordagem de recomendação via elicitación e estimativa de preferência contextual. Adaptado de Adomavicius; Tuzhilin (2011)	29
Figura 4.1: Interface exibindo o mapa e a lista de objetos de aprendizagem recomendados ao usuário.....	35
Figura 4.2: Interface exibindo o mapa e os detalhes do objeto de aprendizagem selecionado pelo usuário.....	35
Figura 4.3: Visão sumarizada do modelo de contexto proposto com suas quatro dimensões de informação	40
Figura 4.4: Sub-dimensão contextual de informações do perfil acadêmico do usuário .	41
Figura 4.5: Sub-dimensão contextual de informações do perfil geral do usuário	42
Figura 4.6: Dimensão contextual de informações do ambiente (localização).....	44
Figura 4.7: Dimensão contextual de informações dos recursos computacionais	44
Figura 4.8: Dimensão contextual de informações dos objetos de aprendizagem	45
Figura 4.9: Estrutura hierárquica dos conceitos descritos no modelo IEEE LOM.....	47
Figura 4.10: Estrutura hierárquica dos conceitos descritos na ontologia LUBM de Guo, Pan, Heflin (2005)	48
Figura 4.11: Estrutura hierárquica dos conceitos descritos na ontologia de perfil do usuário de Golemati et al. (2007)	49
Figura 4.12: Arquitetura da Solução	50
Figura 5.1: Visão geral do funcionamento da solução	55
Figura 5.2: Ontologia geral instanciada segundo o cenário descrito	61
Figura 5.3: Ontologia instanciada após a execução da regra de recomendação SWRL. 62	
Figura 5.4: Resultado gerado pelo motor de inferência Pellet ao raciocinar sobre a regra SWRL instanciada	62

LISTA DE TABELAS

Tabela 5.1: Notações de linguagem descritiva	56
Tabela 5.2: Comparação de motores de inferência (SIM representa característica suportada, NÃO representa característica não suportada, S/N representa necessidade de melhor explicação)	58

RESUMO

Campus universitários são ambientes compostos de recursos e pessoas que utilizam os tais. Um dos principais recursos utilizados pela comunidade de um campus são os objetos de aprendizagem. Tais objetos existem de maneira abundante, espalhados no ambiente ou concentrados em um único local. Entretanto, a abundância desses objetos faz com que uma pessoa sinta-se cognitivamente cansada ao ter que analisar vários objetos e selecionar apenas alguns. Esse cansaço cognitivo acaba levando a pessoa a escolher um conjunto de objetos de aprendizagem que não satisfarão suas necessidades e interesses da melhor maneira possível.

A computação evoluiu de grandes mainframes a pequenos computadores espalhados em um ambiente. Hoje é possível a existência de ambientes pervasivos, onde os recursos computacionais estão sempre presentes e agindo de forma invisível ao usuário. Tais ambientes tornam possível o acompanhamento das atividades do usuário, provendo informações contextuais que podem ser utilizadas para ajudar a seleção dos melhores recursos (ex. objetos de aprendizagem, restaurantes, salas de aula) à determinada pessoa.

A localização é uma informação contextual de grande importância na seleção de tais recursos. Tal informação pode ser facilmente obtida através do sinal de GPS do dispositivo móvel de um usuário e utilizada em conjunto com os interesses do usuário para recomendar os recursos próximos que melhor atenderão ao mesmo.

Neste contexto este trabalho descreve uma abordagem para recomendar objetos de aprendizagem físicos ou virtuais que estejam relacionados aos prédios próximos a atual localização do usuário. Para executar tal tarefa é descrito um sistema de recomendação que utiliza a informação de localização, obtida através do dispositivo móvel do usuário, combinada à informações do perfil do usuário, dos objetos de aprendizagem relacionados aos prédios e informações tecnológicas do dispositivo para instanciar um modelo ontológico de contexto.

Após instanciado o modelo são utilizadas regras semânticas, escritas em forma de antecedente e conseqüente, que fazem uma correspondência entre os interesses do usuário e o domínio de conhecimento do objeto de aprendizagem para filtrar os objetos próximos ao usuário. De posse desses objetos recomendados o sistema os apresenta em uma interface adaptativa que mostra a localização tanto dos objetos quanto do usuário.

Para validar a abordagem apresentada é desenvolvido um estudo de caso onde as regras semânticas de recomendação são executadas sobre o modelo ontológico desenvolvido. O resultado gerado por tais regras é um conjunto de pares (usuário, objeto de aprendizagem recomendado) e prova a validade da abordagem.

Palavras-Chave: Recomendação de Recursos, Ambientes Pervasivos, Sensibilidade ao Contexto, Interfaces Adaptativas, Modelos Ontológicos.

Context-Aware Adaptive Recommendation of Resources for Mobile Users in a University Campus

ABSTRACT

University campus are environments composed of resources and people who use them. One of the main resources used by a campus community are learning objects. Such objects are abundantly even scattered in the environment or concentrated in one location. However the abundance of such objects makes a person feel cognitively tired when having to analyze various objects and select just a few of them. This cognitive fatigue eventually leads the person to choose a set of learning objects that do not meet their needs and interests in the best possible way.

Computing has evolved from large mainframe to small computers scattered in an environment. Today it is possible the existence of pervasive environments where computational resources are always present and acting in a manner invisible to the user. Such environments make it possible to monitor user activities, providing contextual information that can be used to help select the best resources (e.g. learning objects, restaurants, classrooms) to a particular person.

The location is a contextual information of great importance in the selection of such resources. Such information can be easily obtained through the GPS signal from a mobile device and used with the user's interests to recommend the nearby resources that best attend his needs and interests.

In this context, this work describes an approach to recommend physical or virtual learning objects that are related to buildings near the user's current location. To accomplish such a task we described a recommender system that uses the location information, obtained through the user's mobile device, combined with information from the user's profile, learning objects related to buildings and technological information from the device to instantiate an ontological context model.

Once the model is instantiated we used semantic rules, written in the form of antecedent and consequent, to make a match between the user's interests and the knowledge domain of the learning object in order filter the user's nearby objects. With such recommended objects, the system presents them in an adaptive interface that shows both the object and the user location.

To validate the presented approach we developed a case study where the recommendation semantic rules are executed on the developed ontological model. The income generated by such rules is a set of pairs (user, recommended learning object) and proves the validity of the approach.

Keywords: Resources Recommendation, Pervasive Environments, Context Awareness, Adaptive Interfaces, Ontological Models.

1 INTRODUÇÃO

A evolução da tecnologia tem transformado a forma que as pessoas interagem com computadores. Nos primórdios da computação grandes mainframes ocupavam salas inteiras e tinham seu uso compartilhado. Os paradigmas foram mudando e o computador foi ficando cada vez menor e mais pessoal. Hoje uma única pessoa possui e interage com vários computadores durante o seu dia, por exemplo, um *smartphone*, um computador pessoal no trabalho e um *notebook* em casa.

A diminuição física e o barateamento dos recursos computacionais, como memória e processador, foram alguns dos fatores responsáveis por essa mudança de paradigmas e pelo aumento do número de computadores disponíveis por pessoa. O telefone celular é um exemplo de dispositivo que evoluiu e se popularizou devido a tais fatores. Hoje existem *smartphones* que possuem sistemas operacionais complexos e são dotados de processador e memória comparáveis ou até superiores aos presentes em um computador pessoal.

A popularização desses dispositivos no Brasil é mostrada através de uma pesquisa realizada pelo Instituto Brasileiro de Geografia e Estatística (IBGE) em 2011, esta pesquisa apontou que no período de 2005 a 2011 houve um aumento de 59,7 milhões de pessoas no contingente de detentores de celular para uso pessoal. Os números tiveram um aumento de 107,2% no período da pesquisa, já que em 2005 existiam 55,7 milhões de pessoas que possuíam um aparelho celular para uso pessoal o que representa 36,6% da população do país, já em 2011 esse número passou para 115,4 milhões de pessoas, o que representa 69,1% da população do país (INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA, 2011). A evolução desse crescimento é exibida na Figura 1.1, que além dos números absolutos (dados em milhões de pessoas) mostra também a variação percentual desse crescimento entre um ano e outro.

Outra pesquisa realizada pelo Instituto Brasileiro de Opinião Pública e Estatística (IBOPE) aponta que 16% dos brasileiros já possuem celular do tipo *smartphone* e que em 2013 houve um aumento de 16% nas vendas desse tipo de aparelho (INSTITUTO BRASILEIRO DE OPINIÃO PÚBLICA E ESTATÍSTICA, 2013).

A popularização dos celulares e *smartphones* não se restringe somente ao Brasil, dados publicados pela União Internacional de Telecomunicações mostram que em 2013 a estimativa era que o número de telefones celulares alcancem o número de habitantes do planeta (aproximadamente 7 bilhões). A mesma pesquisa prevê que em 2013 a telefonia celular se difundiria mais de 96% mundialmente, sendo 89% nos países em desenvolvimento e 128% nos industrializados (INTERNATIONAL TELECOMMUNICATION UNION, 2012).

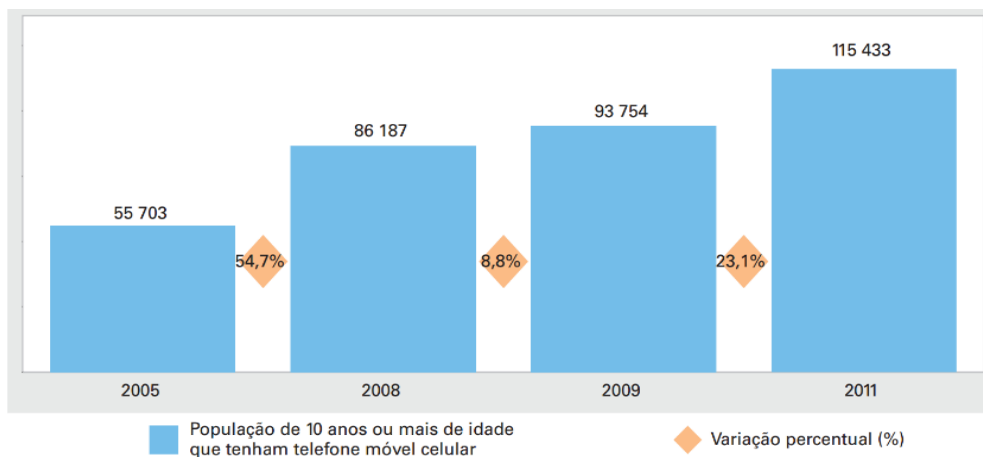


Figura 1.1: Pessoas e variação do número de pessoas de 10 anos ou mais de idade que tinham telefone móvel celular para uso pessoal - Brasil - 2005/2011 (INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA, 2011).

Esses números mostram um interesse crescente das pessoas no uso de *smartphones*. Este crescente interesse também é refletido na área acadêmica, hoje existem diversas conferências (ex. MOBICOM, MOBIHOC, MobiSys) e periódicos (ex. *Pervasive and Mobile Computing*, *Mobile Networks and Applications*, *International Journal of Wireless and Mobile Computing*) especializados em publicações de pesquisas com foco em dispositivos móveis, dentre eles o celular.

A área da computação pervasiva, por exemplo, foi uma das áreas do conhecimento que rapidamente adotou a utilização de dispositivos móveis nos experimentos e validações de suas propostas. Na Seção 3.3 são mostrados alguns trabalhos que utilizam dispositivos móveis para experimentos e validações de propostas em ambientes pervasivos. Ambientes pervasivos são ambientes onde os recursos computacionais estão integrados de maneira que os mesmos se tornam invisíveis ao usuário que pode utilizar um dispositivo como um portal que o leva a ter acesso aos serviços oferecidos por tais recursos invisíveis do ambiente, outrossim estes recursos computacionais estão disponíveis a todo o momento para o usuário dentro do ambiente (BANAVAR et al., 2000).

Um exemplo de ambiente que pode se tornar pervasivo é o campus universitário. Este ambiente é formado por vários recursos do mundo físico (ex. restaurantes, bibliotecas, salas de aula, computadores, livros), do mundo virtual (ex. lâminas de uma aula, artigos científicos, softwares, livros eletrônicos) e de pessoas que utilizam esses recursos. Devido à grande quantidade de recursos espalhados neste ambiente, escolher um conjunto destes que mais se adequem às necessidades do usuário é uma tarefa difícil.

Objetos de aprendizagem são recursos abundantes e de grande interesse em universidades, mesmo assim apesar de todo o esforço investido por estas a fim de organizar e disseminar conhecimento sobre objetos de aprendizagem, não é difícil encontrar pessoas utilizando livros, lâminas de aula e outros tipos de objetos que não atenderão seus anseios e interesses.

Uma das razões que levam as pessoas a escolher um conjunto de objetos de aprendizagem não compatível com seus interesses é que muitas vezes as universidades apresentam uma quantidade muito grande desses recursos espalhados no ambiente ou até mesmo concentrados em uma única localização. Desta forma, a escolha de um recurso a

ser utilizado torna-se uma tarefa extremamente difícil, já que a pessoa fica cognitivamente sobrecarregada por ter que analisar e filtrar os objetos que possam ser de seu interesse. Esta sobrecarga cognitiva acaba levando a pessoa a escolher um conjunto de recursos que não atenderá suas necessidades da melhor maneira possível.

Tecnologias existentes provêm maneiras de rastrear e coletar as informações geradas pelas pessoas dentro de uma universidade. É possível, por exemplo, manter um monitoramento da localização de uma pessoa e saber quais computadores foram utilizados pela mesma e assim tal informação pode ser utilizada para auxiliar na seleção do melhor conjunto de computadores próximos à sua localização.

Uma das alternativas para prover uma solução ao problema da seleção de recursos em ambientes pervasivos é a utilização de sistemas de recomendação. Entretanto para que um sistema de recomendação seja eficaz em recomendar recursos de um ambiente ele deve ser capaz de lidar com informações diferentes das manipuladas pelos sistemas tradicionais, já que ele deve ter ciência dos recursos e do relacionamento deles com o usuário no ambiente.

Esta ciência do relacionamento entre o usuário e os recursos disponíveis é definida como ciência do contexto ou sensibilidade ao contexto - termo adotado neste trabalho. Dey (2001) define que um sistema só é sensível ao contexto se ele usa tal informação para prover informações e/ou serviços relevantes ao usuário.

Um exemplo de informação que pode ser utilizada para mostrar o relacionamento do usuário com os recursos do ambiente é a localização. Através da ciência da localização do usuário e dos recursos, um sistema pode prover recomendações mais relevantes a atual situação de um usuário. De forma que, ao invés de recomendar todos os recursos que possam ser de interesse do usuário, independentemente de sua localização, o sistema recomenda todos os recursos de interesse que estão próximos à atual localização do mesmo.

Note que a informação de localização é utilizada para filtrar as recomendações, o que as torna mais adequadas à atual situação do usuário (ex. próximo a biblioteca) e o sistema eficaz para recomendar recursos próximos em um ambiente pervasivo. Por conta disto, a localização é utilizada neste trabalho como informação de contexto responsável por filtrar recursos de interesse que estejam próximos ao usuário.

Além de utilizar a informação de contexto, um sistema apto para recomendar recursos em um ambiente pervasivo deve agir também de maneira pervasiva, ou seja, deve atualizar suas recomendações de maneira constante e transparente para o usuário. Desta forma, se o usuário mudar sua localização o sistema deve atualizar as recomendações de recursos de acordo com a nova localização do mesmo.

Após selecionar um conjunto de recursos a ser recomendado, deve-se pensar na maneira que tais recursos serão apresentados. Uma interface que se adapta às recomendações providas pelo sistema pode facilitar a utilização do sistema e por isso sua adoção. Brusilovsky (2001) define que um sistema que se comporta de maneira diferente para diferentes usuários no mesmo contexto é chamado de sistema adaptativo.

Este trabalho portanto, tem por objetivo propor uma abordagem sensível ao contexto e de interface adaptativa, a ser utilizada em um dispositivo móvel, para recomendar objetos de aprendizagem próximos e interessantes ao usuário, sendo este um membro da comunidade acadêmica em um campus universitário. Para atingir tal objetivo, este trabalho desenvolve e apresenta nos próximos capítulos um modelo ontológico de

contexto, um sistema de recomendação, uma interface adaptativa e um filtro de objetos de aprendizagem construído através de regras lógicas utilizadas para selecionar um conjunto de objetos de aprendizagem de interesse do usuário.

A abordagem proposta é apresentada no restante deste trabalho, que está organizado da seguinte maneira. No Capítulo 2, são apresentados conceitos a respeito de contexto e modelagem de contexto, no mesmo capítulo é apresentado um estudo de trabalhos relacionados à modelagem de contexto em ambientes pervasivos. No Capítulo 3, é apresentado um estudo geral sobre a área de sistemas de recomendação sensíveis ao contexto, onde são apresentados conceitos e trabalhos relacionados ao tema. No Capítulo 4, é apresentada a abordagem sensível ao contexto composta da especificação da solução, da interface e da arquitetura da solução proposta além do modelo ontológico de contexto a ser utilizado para recomendar objetos de aprendizagem no campus. No Capítulo 5 é apresentado o estudo de caso para validação da abordagem e do modelo, neste capítulo são apresentados um cenário de uso, um estudo comparativo entre motores de inferência e as regras semânticas utilizadas para gerar recomendações. Por fim no, Capítulo 6, são apresentadas as conclusões, resultados alcançados e trabalhos futuros.

2 MODELAGEM DE CONTEXTO

A informação de contexto é um dos elementos de fundamental importância utilizados na computação pervasiva, pois é através dela tenta-se entender o relacionamento entre a computação e o contexto onde ela está inserida. Um exemplo direto que mostra a importância dessa informação é que ao mover um computador para fora da mesa de trabalho de repente é necessário acompanhar para onde ele se foi (DOURISH, 2004).

Dey (2001, p. 2) propõe uma das definições de contexto mais utilizadas na literatura, e adotada neste trabalho, nela é definido que:

Contexto é qualquer informação que pode ser utilizada para caracterizar a situação de uma entidade. Uma entidade é uma pessoa, lugar ou objeto que é considerada relevante para a interação entre o usuário e a aplicação, incluindo o próprio usuário e a aplicação.

O trabalho de Dourish (2004) classifica o contexto segundo sua origem, na computação pervasiva, em duas vertentes. Na primeira, o contexto é encarado como um problema representacional, da mesma forma que sistemas de software são representacionais, por causa disso uma preocupação principal é como o contexto pode ser codificado e representado. Quatro suposições baseiam esta vertente:

- Contexto é uma forma de informação, ou seja, é algo que pode ser conhecido e por isso codificado e representado.
- Contexto é delineável. Para um dado conjunto de aplicações ou requisitos, pode-se definir antecipadamente o que irá contar como contexto de atividades que a aplicação suporta e executa.
- Contexto é estável. Apesar dos elementos da representação de contexto variarem de aplicação para aplicação, eles não variam de instância para instância de uma atividade ou um evento. A determinação da relevância de qualquer potencial elemento de contexto pode ser realizada uma única vez.
- Contexto e atividade são separáveis. As atividades ocorrem dentro de um contexto. O contexto, por sua vez, descreve características do ambiente onde a atividade ocorre, mas são separadas da atividade. Uma pessoa pode estar conversando com outra em uma localização, então a conversa é a atividade da pessoa e a localização é um dos aspectos do contexto.

Essas quatro suposições baseiam a noção de contexto na área de pesquisa em computação pervasiva, independente da definição de contexto adotada. A ideia que o contexto consiste de um conjunto de características de um ambiente que circunda uma atividade genérica, e que essas características podem ser codificadas e tornarem-se disponíveis a um sistema de software junto com uma codificação da própria atividade, é uma suposição comum em vários sistemas.

Na segunda vertente o contexto é visto como um problema interacional, segundo esta visão o tipo de coisa que pode ser modelada pelas quatro suposições da vertente anterior não é o tipo de coisa que o contexto é. A principal diferença aqui é que o contexto não é visto como algo estático que pode ser observado e previsto, pelo contrário, ele depende da interação do usuário com o ambiente. Então, nesta visão, contexto não é algo que descreve um cenário e sim algo que as pessoas fazem. É uma realização, em vez de uma observação, um resultado, ao invés de uma premissa. As quatro suposições que baseiam esta vertente são:

- Contextualidade é uma propriedade relacional que acontece entre objetos ou atividades, ao invés de ser visto como uma informação. Algo não é mais considerado ou não como contexto, ao invés disso esse algo pode ser, ou não, contextualmente relevante para uma determinada atividade.
- O escopo das características contextuais são definidos dinamicamente, ao invés de considerar que o contexto pode ser delineado e definido antecipadamente.
- Contexto é uma propriedade ocasionada, ao invés de considerar o contexto como algo estático, relevante para um cenário particular, instâncias particulares de uma ação e participantes particulares desta ação.
- Contexto surge da atividade, ao invés de considerar contexto e atividade como duas entidades separadas. O contexto não está simplesmente “vagando por aí”, mas é ativamente produzido e mantido no curso da atividade em questão.

Neste trabalho foi adotada a classificação de contexto como um problema representacional. Uma das vantagens ao encarar o contexto como este tipo de problema é que os elementos que irão participar do mesmo (ex. localização, data, preferências) têm um escopo mais limitado e podem ser previamente definidos, facilitando a modelagem das informações de contexto.

Entretanto, na vertente que encara o contexto como um problema interacional o cenário é outro, pois neste caso os elementos que compõe o contexto são variáveis. Por conta disso, dois usuários podem executar a mesma atividade, no mesmo momento e ainda assim possuem elementos de contexto diferentes. Devido a tais características, o contexto neste trabalho é encarado como um problema representacional.

2.1 Técnicas de Modelagem de Contexto

Segundo Bettini et al. (2010) é importante que se utilize uma técnica de modelagem durante o desenvolvimento do modelo de contexto já que tais técnicas são suportadas por adequados métodos de engenharia de software, que têm o intuito de reduzir a complexidade das aplicações geradas e melhorar a manutenibilidade e a capacidade evolutiva de tais aplicações.

Os autores ainda classificam as técnicas para modelagem de contexto em: i) baseadas em papéis objeto, utilizam linguagem *Context Modelling Language* (CML); ii) baseadas em modelos espaciais, utilizam informações de localização; iii) baseadas em ontologias, utilizam ontologias. Cada uma delas é melhor explicada nas próximas subseções.

2.1.1 Baseadas em papéis objeto

Originada através das tentativas de criar modelos suficientemente formais para apoiar processamentos de consultas e raciocínio, bem como para prover construtos adequados para uso em tarefas de engenharia de software. Uma das principais linguagens utilizadas nesta técnica é a CML, que teve suas origens nas técnicas para modelagem de bancos de dados.

A linguagem provê uma notação gráfica projetada para auxiliar o engenheiro de software na análise e especificação formal dos requisitos de uma aplicação sensível ao contexto. Apesar de utilizar notação gráfica, a formalidade da linguagem torna possível o mapeamento direto de um modelo baseado em CML para um sistema de gerenciamento de contexto que pode ser populado com informações de contexto e consultado por aplicações sensíveis ao contexto.

Contudo uma desvantagem desta linguagem é que ela apresenta um modelo de informação “raso” já que qualquer tipo de informação de contexto é representada da mesma maneira. Por conta disso, se os elementos de contexto apresentam-se de forma hierárquica, ou existe algum elemento de particular importância (como é a localização em modelos espaciais) o uso da linguagem não é apropriado.

2.1.2 Baseadas em modelos espaciais

Baseadas em modelos espaciais: a informação espacial é importante em várias aplicações sensíveis ao contexto, tal informação encontra-se presente na definição de contexto de Dey (2001, p. 2) mostrada na seção anterior “*Uma entidade é uma pessoa, lugar ou objeto que é considerada relevante...*”. As pessoas tendem inclusive a associar certas situações à informação de localização, por exemplo, uma pergunta frequente feita em ligações para celulares é: “Onde você está?”.

A maioria dos modelos espaciais tem origens similares aos modelos de papéis objeto, ou seja, na tentativa de criar modelos de contexto formais e prover construtos úteis à tarefas de engenharia de software. As informações de localização gerenciadas por tais modelos podem estar associadas à localização de entidades físicas (do mundo real) ou entidades não físicas (do mundo virtual, ou metáforas).

Os sistemas de coordenadas suportados por sistemas que utilizam modelos espaciais podem ser: i) Coordenadas geométricas, representa pontos ou áreas no espaço métrico (ex. latitude longitude, altitude), ou ii) Coordenadas simbólicas, são representadas por um identificador (ex. número da sala, ID de um celular em uma rede 3G) e ao contrário das coordenadas geométricas, não existe relação espacial.

2.1.3 Baseadas em ontologias

Segundo Gruber (1993), ontologias são uma “especificação formal e explícita de uma conceituação compartilhada”, é uma conceituação porque refere-se a um modelo abstrato do mundo real, é explícita porque os tipos de conceitos utilizados e suas restrições de uso são explicitamente definidos, é formal porque pode ser entendida por máquina e é compartilhada porque captura um conhecimento consensual, aceito por um grupo.

A linguagem mais utilizada para construção de ontologias é a *Web Ontology Language* (OWL) (BECHHOFER et al., 2004), que em sua versão DL e Lite possibilitam o uso de motores de inferência que utilizam lógica descritiva e são capazes de checar a consistência do modelo desenvolvido, inferir novos relacionamentos entre classes e descobrir conhecimentos anteriormente implícitos.

A linguagem OWL ainda permite a modelagem de um domínio específico através da definição de classes, indivíduos, características de indivíduos (propriedades de tipo de dados) e relacionamentos entre indivíduos (propriedades objeto).

Ontologias podem ser utilizadas para capturar o conhecimento disponível no mundo e transformá-lo em um modelo compreendido por máquina. Por conta disto as ontologias podem então ser utilizadas na modelagem de informações de contexto. Algumas vantagens que modelos de contexto têm ao explorar o poder de representação e raciocínio das ontologias são:

- A expressividade da linguagem é utilizada para descrever dados complexos de contexto que não poderiam ser representados por linguagens mais simples;
- Provêm semântica formal aos dados de contexto, e isto torna possível o compartilhamento e/ou integração de dados de contexto entre diferentes fontes;
- As ferramentas de raciocínio disponíveis podem ser utilizadas tanto para checar a consistência das ontologias como para reconhecer que um determinado conjunto de instâncias de dados básicos de contexto e seus relacionamentos revelam a presença de uma caracterização de contexto mais abstrata (ex. a atividade do usuário pode ser reconhecida automaticamente).

Modelos de contexto que utilizam ontologias são adequados para compartilhamento de conhecimento, já que eles provêm uma especificação formal de dados de contexto semânticos. Segundo Bettini et al. (2010) esta característica é particularmente importante em ambientes móveis e pervasivos, onde entidades diferentes devem interagir com o intuito de compartilhar informações de contexto do usuário.

Devidos as características apresentadas acima, o uso de ontologias mostrou-se a melhor opção para modelar as informações de contexto manipuladas neste trabalho. Por conta disto, nas próximas seções serão apresentados alguns trabalhos relacionados e posteriormente o modelo ontológico de contexto desenvolvido.

2.2 Trabalhos Relacionados

São encontrados na literatura vários trabalhos que utilizam ontologias para modelar o contexto com o intuito de adaptar o comportamento de um sistema. Tais trabalhos podem ser categorizados conforme o objetivo dos modelos: i) de alto nível e ii) aplicados. Os modelos de alto nível têm por objetivo dar suporte à construção dos modelos aplicados, por isso são dotados de conceitos mais genéricos e que podem ser utilizados por várias aplicações de um mesmo domínio. Os modelos aplicados, em contrapartida, são mais específicos e dificilmente são utilizados em sua totalidade por várias aplicações.

2.2.1 Modelos de Alto Nível

O trabalho de Chen et al. (2004), propõe um modelo ontológico de contexto de alto nível batizado de *Standard Ontology for Ubiquitous and Pervasive Applications* (SOUPA), construída em OWL a ontologia utiliza informações definidas em uma série de outras ontologias com o intuito de dar suporte agentes inteligentes (com suas crenças, desejos e intenções) a serem utilizados em ambientes pervasivos. A ontologia é dotada de dois conjuntos distintos de ontologias relacionados, o *SOUPA Core* (de alto nível) e o *SOUPA Extension* (aplicado).

As ontologias pertencentes ao *SOUPA Core* tentam definir informações genéricas que serão utilizados por diferentes aplicações pervasivas, já as ontologias pertencentes ao conjunto *SOUPA Extension* são extensões do *SOUPA Core* e definem informações a serem utilizadas por tipos de aplicações específicas além de prover exemplos para futuras extensões. Algumas categorias de informação manipuladas pelas ontologias do conjunto *Core* são relativas à pessoa, políticas e ações, agentes, tempo, espaço e eventos. As categorias manipuladas pelas ontologias do conjunto *Extension* são relativas à reuniões e agenda, documentos e documentos digitais, captura de imagem e localização.

O trabalho de Wang et al. (2004), propõe um modelo ontológico de contexto de alto nível batizado de *CONtext ONtology* (CONON), construída em OWL a ontologia modela quatro conceitos de alto nível principais, são eles, Pessoa, Localização, Atividade e Entidades Computacionais, dos quais podem derivar conceitos mais específicos. O modelo é dividido em ontologia superior e ontologias específicas de domínio, entretanto os autores não fornecem detalhes a respeito das ontologias específicas, focando o trabalho na ontologia superior. A Figura 2.1 apresenta parcialmente a ontologia superior definida e seu relacionamento com outras ontologias específicas de domínio. Além do modelo ontológico os autores apresentam uma abordagem que utiliza lógica de primeira ordem para raciocinar sobre o modelo e assim extrair conhecimento a partir da ontologia instanciada.

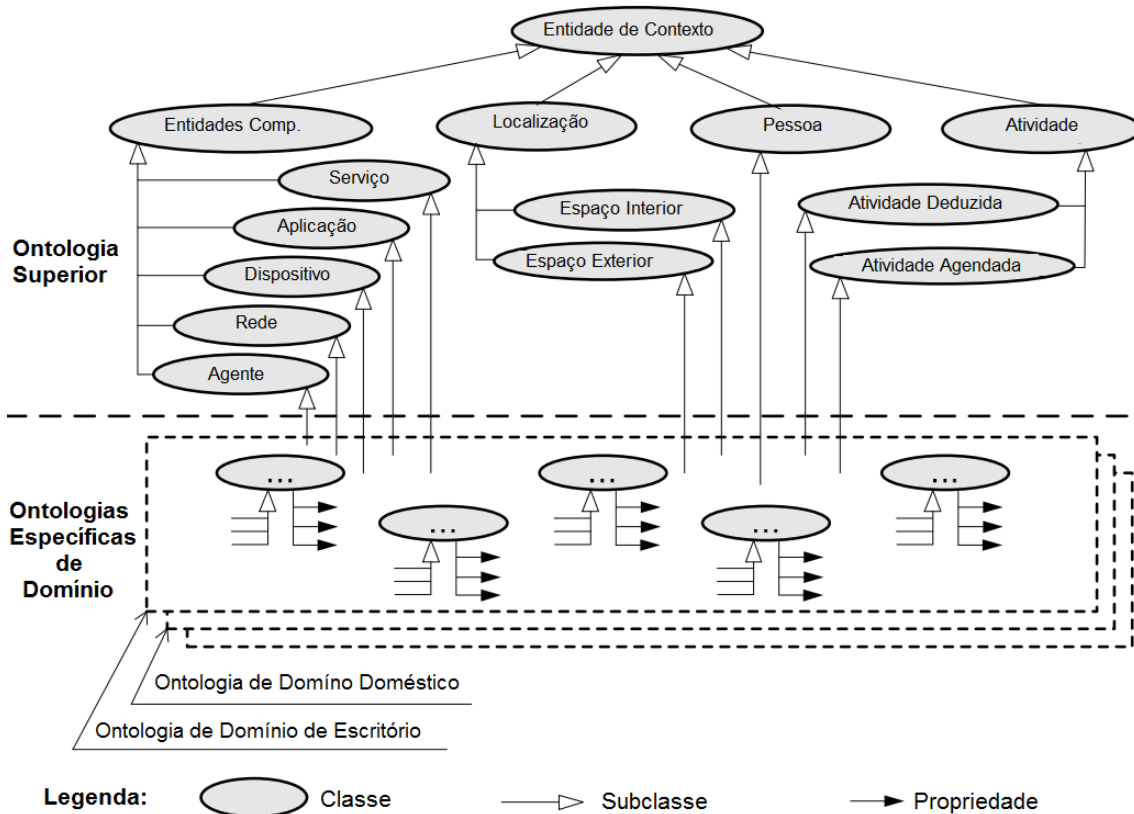


Figura 2.1: Definição parcial da ontologia superior CONON. Adaptado de Wang et al. (2004)

Outro modelo ontológico de contexto de alto nível foi proposto por Hong e Cho (2008), batizado de *Context-Aware Learning ONTology* (CALA-ONT) esta ontologia é bem semelhante à apresentada anteriormente por Wang et al. (2004) tem por objetivo

modelar o contexto de um ambiente de aprendizagem pervasivo que oferece serviços a seus usuários. A ontologia foi construída em OWL e consiste de quatro conceitos de alto nível principais (Pessoa, Lugar, Atividade, Entidades Computacionais), cada um dos quatro conceitos possui sub conceitos (aplicados) e relacionamentos através de propriedades objeto. A ontologia é instanciada com dados de contexto coletados de sensores presentes no ambiente e utiliza regras em lógica de primeira ordem a fim de oferecer serviços a um usuário em um determinado contexto.

O trabalho de Pantsar-Syvaniemi, Simula e Ovaska (2010) propõe um modelo ontológico de contexto de alto nível, construído em OWL para ser utilizado em ambientes pervasivos. O modelo tem por objetivo apoiar a adaptação do comportamento do sistema de acordo com o contexto do usuário. Os autores definem a separação da informação de contexto em três níveis: i) físico, descreve informações do ambiente físico, tais como tempo, localização, equipamentos de rede; ii) digital, funde as informações vindas do primeiro nível a fim de descrever situações tais como “Acidente próximo”, neste nível o modelo ontológico é instanciado e iii) humano, funde informações do nível anterior para encontrar relacionamentos entre situações. Após obter dados de sensores e instanciar o modelo ontológico o sistema propõe a utilização de agentes inteligentes para raciocinar sobre as informações e adaptar o ambiente. Na Figura 2.2 é mostrado o modelo proposto pelos autores e o relacionamento do mesmo com os níveis de contexto.

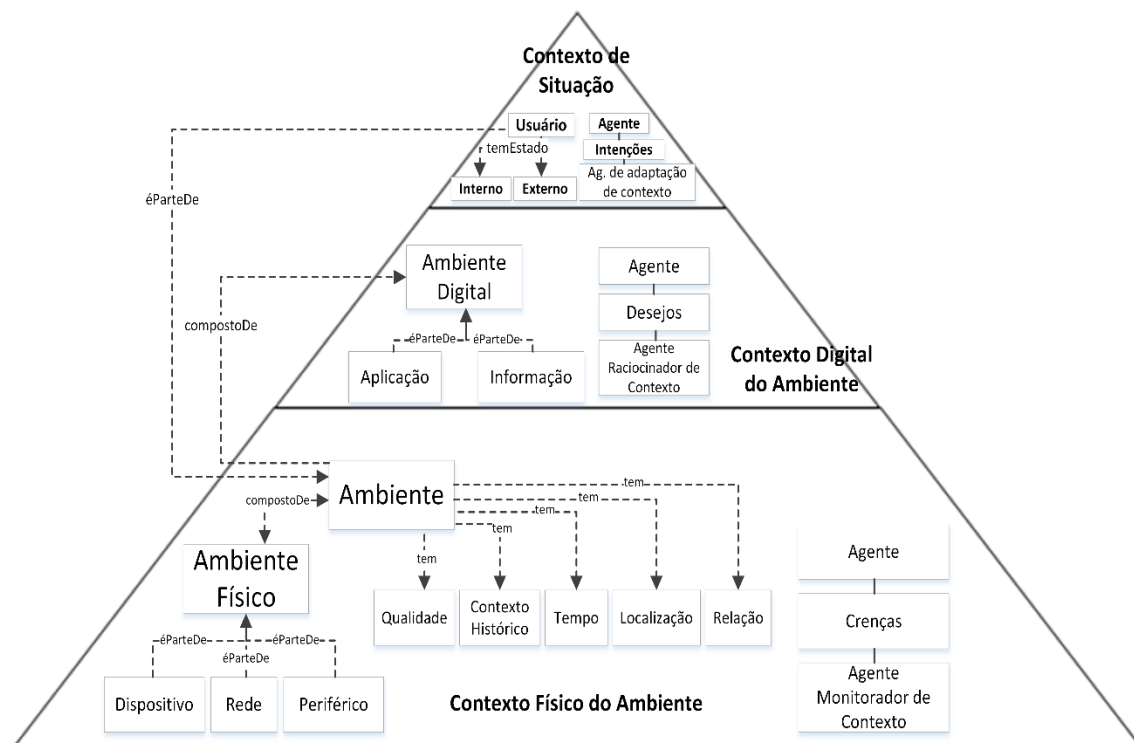


Figura 2.2: Ontologia de Pantsar-Syvaniemi distribuída nos níveis de contexto. Adaptado de Pantsar-Syvaniemi, Simula e Ovaska (2010)

Os quatro trabalhos apresentados nesta subseção preocupam-se em definir um modelo genérico de contexto que possa ser estendido e reutilizado em ambientes pervasivos. Entretanto os autores não se preocupam em modelar objetos não computacionais que serão utilizados pelo usuário. Tais itens têm participação fundamental em atividades suportadas pelo sistema, como por exemplo, o trabalho de Hong e Cho (2008) propõe que o sistema informe ao usuário alguns serviços relevantes em um determinado contexto no

ambiente educacional, porém esses serviços não são previstos na modelagem genérica, ficando por conta das ontologias aplicadas a previsão dos tais.

Uma melhoria que poderia ser executada em tais modelos seria modelagem de elementos não computacionais que fazem parte do contexto do usuário, assim se um sistema tem por intenção, por exemplo, apresentar objetos de aprendizagem relevantes ao contexto do usuário, o modelo genérico daria suporte à modelagem específica de tais objetos de aprendizagem.

2.2.2 Modelos aplicados

Além de modelagens de alto nível são encontrados vários trabalhos que propõem uma modelagem específica de elementos de contexto. No trabalho de Hervás e Bravo (2011) é proposto um modelo de contexto aplicado, construído em OWL que tem por objetivo apoiar a adaptação de interfaces de software através da utilização de informações de ambientes pervasivos. O modelo se baseia em uma ontologia superior, porém o trabalho enfoca-se nas ontologias aplicadas que derivam desta.

Quatro dimensões de informações contextuais compõem tal modelo, são elas, a dimensão do usuário, do dispositivo, do ambiente e do serviço, sendo que as três primeiras dimensões são relacionadas às características de ambientes pervasivos e a última é relacionada às características relativas à visualização de informações. As três primeiras ontologias descrevem: i) o usuário, incluindo sua situação atual (localização, atividades, papéis e objetivos) e seus relacionamentos sociais; ii) os dispositivos relevantes, com suas características, associações e dependências e iii) o ambiente físico, que define a distribuição espacial.

O trabalho de Al-Mekhlafi, Hu e Zheng (2009) propõe um modelo ontológico de contexto aplicado que tem por objetivo ser utilizado por um sistema para recomendar sentenças utilizadas no cotidiano de um estudante da língua chinesa de acordo com seu nível de aprendizado da língua. O modelo é dividido em quatro dimensões principais: i) tempo, descreve o status do sistema (ativado, desativado ou sincronizando); ii) localização, descreve o lugar onde o estudante se encontra no momento; iii) atividade, descreve as atividades planejadas na agenda do estudante e iv) nível, descreve o nível que um estudante pode estar no aprendizado da língua, bem como o nível de uma sentença armazenada no banco de dados. Para realizar as recomendações de sentenças o sistema utiliza as informações de contexto anteriormente descritas e por fim realiza uma correspondência entre o nível do usuário e o nível da sentença a ser recomendada utilizando um motor de inferência para tal.

O trabalho de Yang (2006) propõe um modelo ontológico de contexto aplicado que tem por objetivo apoiar ambientes de aprendizagem pervasivos. Mais especificamente o modelo é utilizado para apoiar a adaptação do comportamento de três softwares utilizados no ambiente utilizado pelos autores o primeiro é um sistema de adaptação e acesso ponto a ponto de conteúdo, o segundo é um sistema de gerenciamento de anotações personalizadas e o terceiro é um sistema multimídia de grupo de discussão em tempo real.

Os autores definem que sensibilidade ao contexto é concebida através de um modelo de interação entre os aprendizes e os professores, por conta disso o modelo ontológico é dividido nas dimensões aprendiz e serviço. A dimensão do aprendiz é composta de informações tais como perfil pessoal, acessibilidade e preferências, perfil calendário, perfil social e perfil de localização. A dimensão do serviço é composta de informações como entrada, saída, pré-condição e efeito da execução do serviço. Além das informações

apresentadas anteriormente ainda são modeladas informações a respeito do contexto que “cerca” o usuário e o serviço, tais como qualidade do serviço de aprendizagem, perfil do ambiente e perfil da capacidade do dispositivo.

Os trabalhos apresentados anteriormente mostram diferentes maneiras de modelar o contexto em ambientes pervasivos, alguns de maneira mais genérica, como é o caso dos modelos de alto nível e outros de maneira mais específica, como é o caso dos modelos aplicados. Nota-se, entretanto, que as informações de contexto manipuladas pelos trabalhos acabam seguindo um padrão, já que todos os trabalhos modelam pelo menos informações do perfil do usuário, informações do ambiente e informações tecnológicas.

2.3 Considerações Finais

O desenvolvimento de um modelo de contexto a ser utilizado em uma aplicação para recomendação de objetos de aprendizagem em uma universidade mostrou-se uma das tarefas mais desafiantes deste trabalho. Através do trabalho de Bettini et al. (2010) foi possível definir qual seria a melhor técnica utilizada para modelar o contexto dadas as características do domínio. Após selecionada a técnica (modelagem através da utilização de ontologias), outro desafio foi descobrir que categorias de informações deveriam ser modeladas.

Através de uma revisão bibliográfica de trabalhos relacionados à área três dimensões de informação mostraram-se básicas para a modelagem do domínio (informações do usuário, informações do ambiente e informações dos recursos computacionais). Entretanto verificou-se que poucos trabalhos importavam-se em modelar informações de recursos não computacionais presentes no ambiente e esta informação é essencial ao sistema de recomendação aqui proposto, uma vez que estes são um dos tipos de item que serão recomendados ao usuário. Outra conclusão que se chegou através dos trabalhos relacionados é que não foram encontrados sistemas de recomendação que utilizassem um modelo de contexto ontológico que contemplasse as características essenciais do domínio a que se propõe a construção do modelo ontológico.

No próximo capítulo é realizada uma breve apresentação da área de sistemas de informação e logo após é apresentada em maiores detalhes a subárea de sistemas de recomendação sensíveis ao contexto. Por fim são apresentados alguns trabalhos relacionados, onde são apontadas algumas possibilidades de extensão dos mesmos.

3 SISTEMAS DE RECOMENDAÇÃO E SENSIBILIDADE AO CONTEXTO

Sistemas de recomendação tradicionais são definidos de forma genérica por Ricci et al., (2011) como técnicas e ferramentas de software usadas para prover sugestões de itens a serem utilizados por um usuário. Esta definição mostra que pelo menos duas dimensões de informação são utilizadas nos sistemas de recomendação tradicionais, são elas, o usuário e o item a ser recomendado. Na abordagem mais comum desse tipo de sistema, conhecida como filtragem colaborativa, as avaliações sobre itens dadas por um usuário são utilizadas com o intuito de gerar recomendações a outro usuário que tenha preferências semelhantes ao primeiro.

Herlocker et al., (2004), identificou as principais tarefas realizadas pelo usuário que podem ser auxiliadas por um sistema de recomendação, são elas:

- Anotação no contexto: dado um determinado contexto, por exemplo uma lista de itens, o sistema destaca alguns itens com base nas preferências de longo prazo do usuário. Um exemplo de sistema que utiliza esta abordagem é o *Tapestry* (GOLDBERG et al., 1992), que foi o primeiro sistema de recomendação desenvolvido. O *Tapestry* recebia uma lista de e-mail corporativos e marcava aqueles que determinado usuário pudesse estar interessado.
- Encontrar bons itens: o sistema provê interfaces que sugerem itens específicos a cada usuário, estes itens são classificados em uma lista e o sistema ainda mostra predições de quanto o usuário gostaria de cada item. Um exemplo de sistema que utilizaria esta abordagem é um sistema de recomendação de filmes que além de recomendá-los ainda os classificaria com base na predição da nota que o usuário daria a cada filme após assisti-lo.
- Encontrar todos os itens bons: a maioria dos sistemas de recomendação sugerem apenas alguns itens interessantes. Alguns usuários estão dispostos a receberem apenas alguns itens interessantes em troca do descarte de muitos itens não interessantes, evitando assim uma sobrecarga cognitiva na análise de muitos itens recomendados. Outros usuário, entretanto, têm interesse em analisar todos os itens interessantes e assim um desafio a este tipo de sistema é evitar falsos positivos. Um domínio que se beneficia deste tipo de abordagem é o domínio jurídico, muitas vezes advogados estão interessados em conhecer todas as possíveis brechas que um documento legal possa ter em relação a legislação vigente.
- Recomendar sequencias: esta tarefa diferencia-se das anteriores pois ao invés de recomendar um item por vez o sistema deve recomendar uma sequência de itens de interesse do usuário. Um exemplo de sistema que se encaixa nesta tarefa são os sistemas que recomendam *playlists* de músicas, outro exemplo de sistema é um recomendador de artigos científicos utilizados para ensinar sobre um determinado campo de estudo.
- Somente navegar: alguns usuários utilizam sistemas de recomendação sem intenção eminente de realizar a atividade fim da recomendação (ex. assistir

um filme, comprar um item), eles utilizam o sistema porque acham a atividade agradável. Para esse tipo de usuário a interface e a facilidade de uso do sistema importam mais que a acurácia dos algoritmos de recomendação.

- Encontrar um recomendador confiável: alguns usuários não acreditam em um recomendador automaticamente, por isso eles testam os sistemas de recomendação para ver se eles irão recomendar, por exemplo, seu filme predileto ou algum filme que ele goste. Alguns usuários chegam a criar perfis falsos com a intenção de ver se as recomendações recebidas serão diferentes. Por causa deste comportamento do usuário, sistemas que sugerem itens que o usuário não tem familiaridade mas provavelmente irá gostar, podem não passar no teste do usuário como sistemas confiáveis.
- Melhorar o perfil: usuários passam a classificar os itens de acordo com seus gostos. Usuários contribuem pois eles acreditam que desta forma a qualidade das recomendações recebidas aumentará.
- Expressar a si mesmo: o usuário classifica os itens do sistema sem intenção de receber recomendações, ele faz isso como uma maneira de expressar suas opiniões e preferências além de se sentir bem fazendo isso. Em alguns sites é possível que além de classificar um item o usuário opine sobre o mesmo, isso faz que ele sinta bem porque contribuiu facilmente com sua opinião. Por causa disso uma estratégia utilizada por sistemas de recomendação é colocar mensagens motivadoras após uma classificação de item, isso estimula o usuário a contribuir mais, melhorando assim a qualidade das recomendações em geral.
- Ajudar a outros: em alguns casos os usuários classificam itens porque acreditam que desta forma a comunidade se beneficiará de sua contribuição. Note que neste caso o usuário classifica para ajudar outras pessoas e não somente para se expressar como no caso anterior.
- Influenciar a outros: alguns usuários querem influenciar outros explicitamente, levando as pessoas, por exemplo a assistirem filmes dos estúdios para os quais eles trabalham, para isso eles sempre classificam os itens que serão influenciados aos outros usuário de forma positiva. Esta tarefa é interessante, pois sistemas devem identificar e tratar esse tipo de influência para que isso não atrapalhe na recomendação.

São encontrados, na literatura, vários sistemas de recomendação que se propõe a auxiliar no cumprimento dessas tarefas. Ricci et al., (2011) propõe uma interessante classificação dos sistemas de recomendação baseando-se na técnica utilizada para recomendar itens.

- Baseado em conteúdo: o sistema aprende a recomendar itens similares aos que o usuário gostou no passado, para isso ele leva em conta características associadas ao itens comparados. Por exemplo se o usuário classificou positivamente um filme do gênero de comédia, o sistema então aprenderá a recomendar outros filmes do mesmo gênero.
- Baseado em filtragem colaborativa: o usuário ativo recebe recomendações de itens que outros usuários com gostos similares classificaram positivamente

no passado. A similaridade de gostos de dois usuários é calculada com base na similaridade do histórico de classificações dos usuários.

- Baseado em demografia: o sistema recomenda itens baseando-se no perfil demográfico do usuário. Assume-se que diferentes recomendações devem ser geradas para diferentes nichos demográficos. Um exemplo comum de uso desta técnica é quando sites da web direcionam seus usuários a páginas traduzidas para um idioma falado em seu país.
- Baseado em conhecimento: o sistema utiliza conhecimento específico de domínio a fim de corresponder características dos itens com necessidades e preferências do usuário ou em último caso, como o item pode ser útil a este. Para realizar essa correspondência uma função de similaridade estima em quanto as necessidades do usuário correspondem às características do item. Por fim os itens mais similares são recomendados. Outra estratégia utilizada por esse tipo de sistema é corresponder as características do item e do usuário através de restrições. Em termos de conhecimento utilizado ambas estratégias são similares, requisitos do usuário são coletados; alternativas para requisitos inconsistentes são propostas a fim de evitar situações sem solução; e resultados são explicados. A principal diferença entre as estratégias é determinada através da maneira que a correspondência usuário e item é calculada. No primeiro caso é utilizada uma função de similaridade para tal e no segundo caso são utilizadas regras explícitas que relacionam o usuário e o item com base no conhecimento prévio do domínio.
- Baseado em comunidade: o sistema recomenda itens com base na preferência dos amigos do usuário. Esse tipo de sistema tem se tornado de grande interesse de estudo ultimamente devido ao advento das redes sociais e também é chamado de sistema de recomendação social já que considera o relacionamento do usuário com seus amigos para recomendar itens.
- Sistemas híbridos: combina duas ou mais técnicas mencionadas acima. Um sistema híbrido que combina as técnicas A e B tenta utilizar as vantagens de A para contornar as desvantagens de B. Por exemplo sistemas que utilizam filtragem colaborativa têm problemas para recomendar itens novos, já que esses nunca foram classificados. Este problema, entretanto, não limita uma abordagem baseada em conteúdo, já que esta considera apenas características intrínsecas do item a ser recomendado.

Dentre todas as técnicas a filtragem colaborativa tornou-se o método mais popular e mais implementado pelos sistemas de recomendação tradicionais (RICCI; ROKACH; SHAPIRA, 2011). Devido a esse fato a próxima subseção explica resumidamente esse método.

3.1 Recomendação Baseada em Filtragem Colaborativa

Sarwar et al., (2001) apresenta a filtragem colaborativa como uma técnica que tem dois objetivos principais, o primeiro é sugerir novos itens a um determinado usuário e o segundo é prever a utilidade de um item ou a classificação que um usuário daria a este. Para atingir tais objetivos a técnica utiliza uma lista de m usuários $\mathcal{U} = \{u_1, u_2, \dots, u_m\}$ e uma lista de n itens $I = \{i_1, i_2, \dots, i_n\}$. Para cada usuário u_i existe uma lista de itens I_{ui} que contém a classificação dada pelo usuário u_i a um item i_j , esta

classificação é expressa através de uma escala numérica e pode ser dada explicitamente pelo usuário ou derivada implicitamente através de logs de comportamento do usuário. A lista pode estar vazia se o usuário ainda não classificou nenhum item. A Figura 3.1 mostra o diagrama esquemático do processo de filtragem colaborativa e os dois objetivos da técnica são melhores explicados a seguir:

- Predição é um valor numérico que representa o quanto o usuário gostaria de determinado item, este item porém não deve estar na lista de itens já classificados para aquele usuário. A predição é dada na mesma escala de valores que um usuário classifica um item (ex. 1 a 5).
- Recomendação é uma lista de itens que o usuário mais gostaria, nesta lista porém não aparecem os itens que o usuário já classificou. Esta interface é também conhecida como as Top-N recomendações.

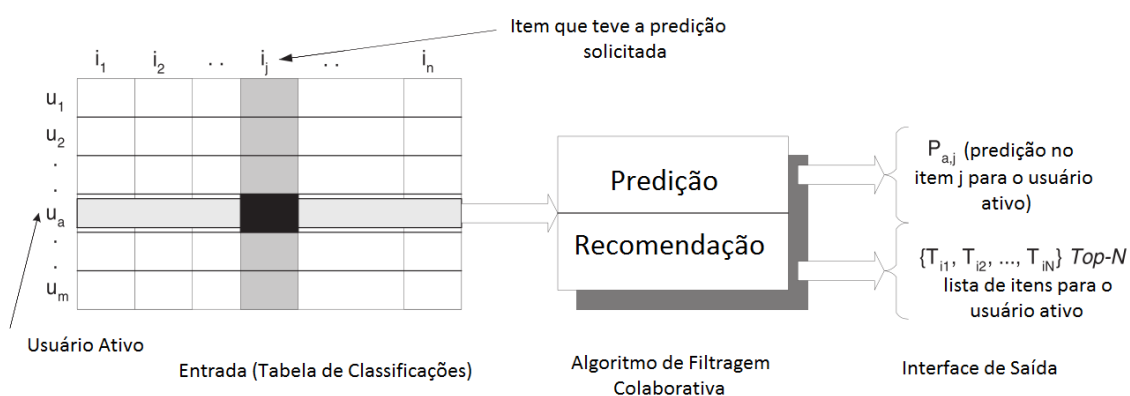


Figura 3.1: O processo de filtragem colaborativa. Adaptado de Sarwar et al., (2001)

Os autores elucidam dois principais problemas relativos aos sistemas de recomendação tradicionais. O primeiro é relativo a falta de escalabilidade das técnicas baseadas no usuário, pois em grandes bancos de dados existem muitos vizinhos (usuários com gostos semelhantes) a serem comparados e alguns vizinhos podem ter um histórico muito grande a ser analisado, causando a redução da escalabilidade da técnica. O segundo problema é relativo às recomendações produzidas que devem ter alta qualidade pois os usuários podem rejeitar um sistema que produz recomendações não confiáveis.

3.2 Recomendação Sensível ao Contexto

É importante notar que algumas vezes a utilidade de um item para um usuário depende de outras variáveis que são chamadas contextuais. Por exemplo, a utilidade de um item pode ser influenciada pelo conhecimento que o usuário tem do domínio (ex. conhecimento básico ou avançado em computadores), ou pode depender da época que a recomendação foi requisitada (ex. locais ensolarados para passar as férias de inverno). Outra possibilidade é o usuário estar interessado em itens próximos a sua atual localização (ex. restaurantes).

Devido a esses fatores Adomavicius et al., (2005) propôs a utilização da dimensão de contexto junto ao usuário e o item a fim de gerar recomendações, surgindo então os sistemas de recomendação sensíveis ao contexto. Os autores propõem a utilização de uma abordagem multidimensional para recomendação. A justificativa é que em alguns domínios, como por exemplo a recomendação de pacotes de viagem, conteúdo personalizado em um site da Web, alguns produtos em uma loja online, ou filmes é

importante incorporar informações contextuais do cenário de decisão do usuário no processo de recomendação.

Na abordagem definida pelos autores, bancos de dados *On-line Analytical Processing* (OLAP) são utilizados para representar as várias dimensões de informações D_i a serem utilizadas durante a recomendação de itens. Cada dimensão é representada por um conjunto de informações, por exemplo, a dimensão do usuário pode conter o nome, endereço, preferências e idade. O produto cartesiano das dimensões $D_1 \times \dots \times D_n$ é chamado de *espaço de recomendação* e a função de previsão da classificação do usuário é definida sobre o espaço de recomendação como mostrado abaixo.

$$R: D_1 \times \dots \times D_n \rightarrow \text{Classificação}$$

A Figura 3.2 mostra o modelo proposto para armazenamento do *espaço de recomendação* e das classificações do usuário, no exemplo é mostrado um cubo OLAP que armazena as dimensões do usuário, do item, do tempo e a classificação dada aos itens referentes ao *espaço de recomendação*.

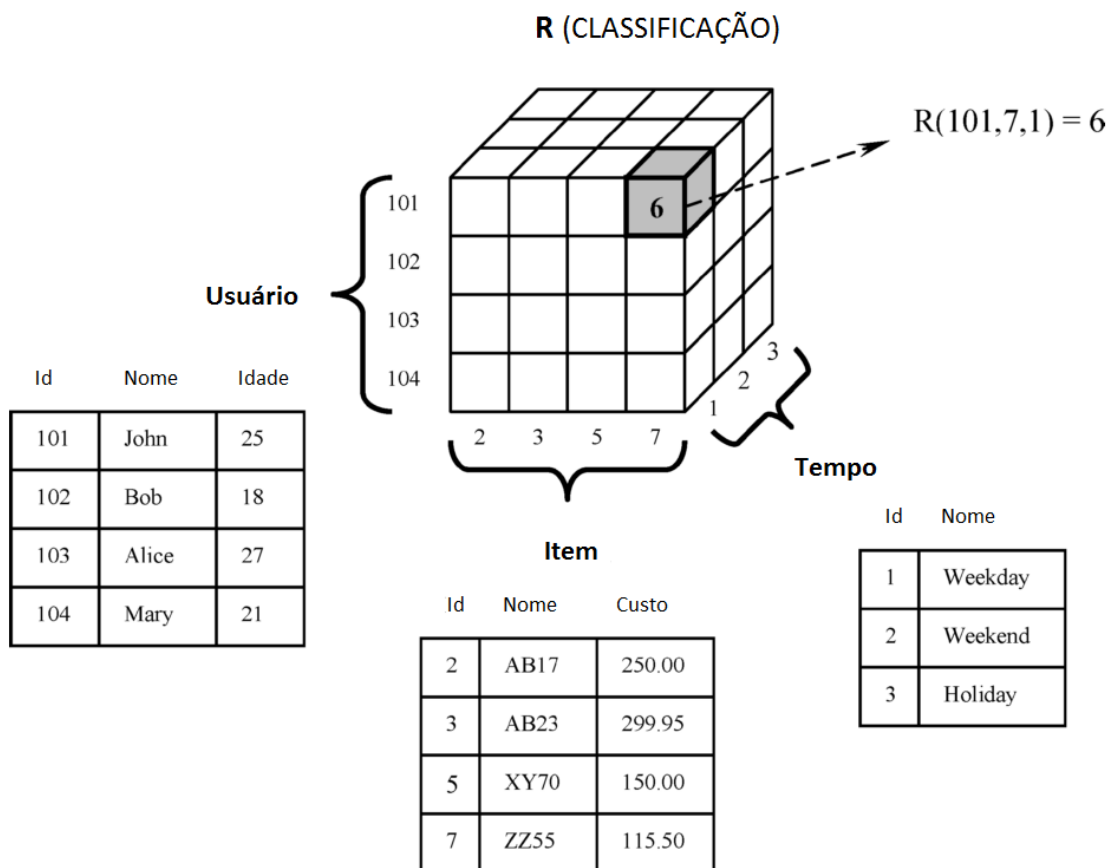


Figura 3.2: Espaço de recomendação do modelo multidimensional para usuário x item x tempo. Adaptado de Adomavicius et al., (2005)

Mais especificamente o problema principal de recomendação é definir quais dimensões de itens D_{i1}, \dots, D_{ik} ($k < n$) devem ser recomendadas para quais determinadas dimensões de usuários D_{j1}, \dots, D_{jl} ($l < n$) de maneira que $\{D_{i1}, \dots, D_{ik}\} \cap \{D_{j1}, \dots, D_{jl}\} = \emptyset$ e que para cada tupla $(d_{j1}, \dots, d_{jl}) \in D_{j1} \times \dots \times D_{jl}$ seja recomendada a tupla $(d_{i1}, \dots, d_{ik}) \in D_{i1} \times \dots \times D_{ik}$ que maximize a classificação $R(d_1, \dots, d_n)$, ou seja,

$$\forall (d_{j_1}, \dots, d_{j_l}) \in D_{j_1} \times \dots \times D_{j_l}, (d_{i_1}, \dots, d_{i_k}) = \operatorname{argmax} R(d'_1, \dots, d'_n)$$

Por exemplo, considere um sistema que recomenda filmes para um usuário e que leva em consideração as dimensões do filme (ex. ID, nome, estúdio, gênero, diretor...), da pessoa (ex. ID do usuário, nome, idade, endereço...), do lugar onde o filme será assistido (ex. um cinema, na TV de casa...), de tempo (ex. dia da semana, período do dia, mês...) e da companhia da pessoa com quem o filme será assistido (ex. sozinho, amigos, namorada, família...). Desta forma a predição da classificação dada pelo usuário a um filme dependerá de onde, quando, como e com quem o filme será visto, por exemplo, filmes diferentes serão recomendados a um determinado usuário se ele estiver na companhia de sua namorada em um cinema num final de semana ou na companhia da família em casa em uma quarta-feira.

Uma importante questão no modelo definido pelos autores é determinar se uma dimensão tem importância significativa na determinação da predição da classificação do usuário. Para isso a predição é executada várias vezes considerando diferentes valores assumidos pela dimensão, logo após, são executados testes estatísticos para determinar se o resultado das predições são significativamente diferentes, se não, a dimensão não deve ser incluída no modelo de recomendação.

As recomendações produzidas por sistemas multidimensionais são diferentes das produzidas por sistemas tradicionais já que nestes existe apenas um tipo de recomendação “recomende os primeiros N itens para o usuário”, enquanto que naqueles as recomendações podem ser entregues de várias formas “melhores combinações N de usuário/tempo para cada item” ou “melhores N itens para cada combinação usuário/tempo” ou ainda “melhores combinações de tempo para cada combinação usuário/item” e assim por diante.

3.2.1 Agregação de informações

Como dito anteriormente o modelo multidimensional proposto por Adomavicius et al., (2005) utiliza bancos de dados OLAP para armazenar as várias dimensões de informação relacionadas ao processo de recomendação de itens. Um dos motivos de utilizar tal modelo de armazenamento de dados é a possibilidade do cruzamento de informações tornando disponível, por exemplo, qual classificação foi atribuída por um usuário a um filme em um determinado local e dia da semana.

Entretanto as dimensões de informações D_i propostas pelos autores requerem uma agregação hierárquica. Um exemplo disso é que um determinado filme armazenado em um cubo OLAP pode ser agregado em subgêneros, que podem ser agregados em gêneros e assim por diante. Essa agregação hierárquica de informações possibilitará aos sistemas de recomendação a habilidade de prover recomendações mais complexas que tratam não apenas de itens individuais mas também de grupos de itens.

Devido a esta capacidade é possível saber não apenas os itens que um usuário gosta, mas também a categoria de tal item, por exemplo, ao invés de obter apenas informações que João gosta do filme chamado O Poderoso Chefão é possível saber que João gosta de filmes do gênero Drama. De maneira similar é possível agregar grupos e assim obter informações como, por exemplo, universitários gostam de filmes de ação. Utilizando uma função de agregação é possível, através da classificação dada ao usuário a cada item, prever a classificação que o usuário daria a um grupo.

3.2.2 Incorporando contexto aos sistemas de recomendação

Segundo Adomavicius e Tuzhilin (2011) existem as abordagens para incorporar informação contextual no processo de recomendação podem ser categorizadas em dois grupos: (i) recomendação via consulta e pesquisa orientada ao contexto, e (ii) recomendação via elicitación e estimativa de preferência contextual.

A abordagem de recomendação via consulta e pesquisa orientada ao contexto (utilizada neste trabalho e melhor detalhada no Capítulo 4) utiliza informações contextuais obtidas diretamente do usuário (ex. humor, interesse, preferências) ou obtidas do ambiente (ex. localização, clima, hora local), a fim de executar consultas em um repositório de recursos e apresentar os melhores recursos para o usuário naquele contexto (ex. restaurantes abertos próximos ao usuário).

A abordagem de recomendação via elicitación e estimativa de preferência contextual tenta modelar e aprender as preferências do usuário através da observação da interação deste e de outros usuários com o sistema, ou ainda através do feedback do usuário em vários itens recomendados anteriormente. Esta abordagem usa técnicas de filtragem colaborativa, baseada em conteúdo ou técnicas híbridas para recomendar itens ou ainda aplica várias técnicas inteligentes de mineração de dados ou aprendizado de máquina para análise de dados.

Na abordagem de recomendação via elicitación e estimativa de preferência contextual o contexto pode ser aplicado em vários estágios do processo de recomendação. Por causa disso esta abordagem pode ser subdividida em três diferentes paradigmas explicados a seguir:

- a) Pré-filtragem contextual: neste paradigma a informação contextual é aplicada durante a seleção ou a construção dos dados. Desta forma, a informação contextual é utilizada para selecionar as possíveis notas que o usuário daria a um determinado item e posteriormente qualquer algoritmo tradicional (que considera apenas usuário e item) é aplicado.
- b) Pós-filtragem contextual: neste paradigma de recomendação a informação contextual é inicialmente ignorada e qualquer algoritmo de recomendação tradicional é aplicado aos dados. De posse dos resultados do processo de recomendação a informação contextual é então aplicada.
- c) Modelagem contextual: neste paradigma de recomendação a informação contextual é utilizada diretamente na técnica de modelagem como parte do processo de estimativa de classificação.

A Figura 3.3 mostra uma visão geral dos três paradigmas de incorporação de contexto explicados. Basicamente tais paradigmas são compostos por três elementos: os dados (entrada), um recomendador (função) e uma lista de itens recomendados (saída). Entretanto cada paradigma tem suas peculiaridades, como o paradigma de pré-filtragem que apresenta o elemento adicional de dados contextualizados. O paradigma de pós-filtragem apresenta o elemento adicional de recomendações não contextualizadas. Por fim, o paradigma de modelagem conceitual apresenta um recomendador sensível ao contexto ao invés de um recomendador tradicional, como nos outros dois paradigmas. As letras $U \times I \times C \times R$ correspondem respectivamente às dimensões do usuário \times item \times contexto \times classificação (ratings).

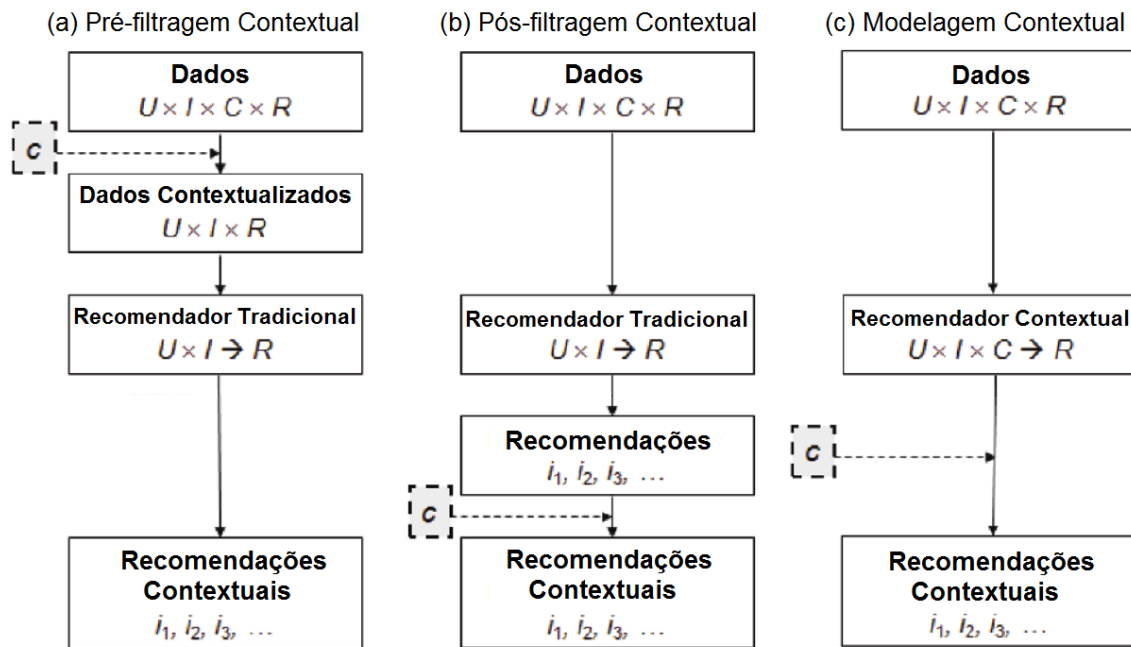


Figura 3.3: Paradigmas para incorporar informação contextual em sistemas que utilizam a abordagem de recomendação via elicitación e estimativa de preferência contextual. Adaptado de Adomavicius; Tuzhilin (2011)

3.3 Trabalhos Relacionados

Um dos primeiros sistemas de recomendação foi desenvolvido por Goldberg et al., (1992) e foi motivado pelo crescente uso de correio eletrônico no início da década de 90. O principal problema causado pela adoção do correio eletrônico é que naquela época não existia um mecanismo eficiente para filtrar as mensagens recebidas por um usuário, por conta disso as pessoas eram inundadas por um enorme fluxo de documentos recebidos.

Tapestry, como foi batizado o sistema, basicamente verificava o conteúdo da mensagem e utilizava o endosso dado por outros usuários, do mesmo escritório, sobre itens (mensagens de e-mail corporativas) para realizar recomendações. Desta forma itens que tivessem conteúdos de interesse e recebessem o endosso de usuários com características semelhantes ao usuário atual eram recomendados. Além da recomendação o sistema provia uma classificação das mensagens por ordem de prioridade. Tal abordagem foi batizada pelos autores com o nome de filtragem colaborativa.

O trabalho de Setten et al., (2004) apresenta um dos primeiros sistemas de recomendação sensíveis ao contexto. A aplicação desenvolvida pelos autores foi batizada de COMPASS, acrônimo de *C*Ontext-aware *M*obile *A*SSistant, e tinha o objetivo de prover informações e serviços de interesse ao contexto de um turista. O desenvolvimento do sistema foi motivado devido à grande quantidade de pontos turísticos presentes em algumas cidades. Por causa dessa grande quantidade as pessoas necessitavam de auxílio na hora de escolher quais pontos deveriam visitar.

O sistema levava em consideração a informação de localização do usuário para mostrar em um mapa, presente no dispositivo móvel do mesmo, a localização de prédios, pessoas e objetos de interesse próximos. Além disso o sistema calculava a previsão de interesse do usuário em relação a um item recomendado. A seleção da técnica utilizada

para calcular a previsão era feita com base em informações do perfil do usuário, do ponto turístico de interesse, descritos em uma ontologia, e informações temporais.

Ambos os sistemas descritos anteriormente forma importantes por serem pioneiros em suas áreas. *Tapestry* como um dos primeiros sistemas de recomendação tradicionais e COMPASS como um dos primeiros sistemas de recomendação sensíveis ao contexto. Uma diferença substancial é percebida entre os dois sistemas, além da inclusão do contexto como dimensão a ser levada em consideração durante a recomendação, é que COMPASS é eficaz para recomendar itens presentes no mundo real e não somente itens virtuais, como e-mails, filmes e músicas.

O contexto foi o elemento que tornou possível gerar recomendações eficazes para elementos do mundo, pois tornou possível, por exemplo, que o sistema recomendasse apenas itens de interesse próximos ao usuário. Este fato causou um interesse crescente por parte da comunidade acadêmica na pesquisa e aplicação de sistemas de recomendação sensíveis ao contexto em ambientes pervasivos. A seguir são mostrados alguns trabalhos relacionados que exemplificam a utilização de tais sistemas tanto em ambientes pervasivos como em ambientes virtuais para recomendar itens no domínio educacional.

O trabalho de Bouzeghoub et al., (2009) que propõe um sistema de recomendação sensível ao contexto e baseado em conhecimento usado para sugerir prédios próximos à localização do usuário, pessoas com mesmos interesses, computadores disponíveis e documentos. Os autores utilizaram uma ontologia e um motor de inferência para construir suas regras de recomendação. Contudo, a abordagem proposta poderia ser expandida se os autores enriquecessem seu modelo ontológico, fazendo-o apresentar mais informações sobre o modelo do usuário, tais como a profissão, papel na universidade e interesses. Além disso poderiam ser apresentadas informações sobre objetos de aprendizagem dada a importância dos tais no domínio acadêmico.

O trabalho de Pernas et al., (2012) apresenta uma rede de ontologias para prover adaptação de interface, de estilo de navegação e uma recomendação, baseada em regras, de conteúdo em sistemas de ensino eletrônico (*e-learning*). A rede é composta por três dimensões que cobrem diferentes domínios no contexto educacional, são eles, o domínio do estudante, contendo informações sobre o perfil do estudante e seu comportamento no ambiente, o domínio de aprendizado, contendo informações sobre os objetos de aprendizagem e os recursos educacionais disponíveis e o domínio tecnológico, contendo informações sobre os dispositivos e lugares que cercam o estudante. Por fim, define uma ontologia de situação, com informações de contexto do aluno em um determinado tempo de início e de fim. Contudo, a proposta dos autores poderia ser expandida se a dimensão de tecnologia fosse dividida, incluindo assim a dimensão de localização ao modelo provendo maiores informações sobre os ambientes ao invés de apenas mostrar o ponto no espaço onde o estudante está localizado.

O trabalho de Wang; Wu, (2011) apresenta um sistema de recomendação sensível ao contexto que utiliza filtragem colaborativa para sugerir materiais didáticos a estudantes. Através de um PDA ou um *smartphone* o usuário realiza leituras de Identificação por Radiofrequência (RFID) em objetos de aprendizagem do mundo real então o sistema sugere materiais didáticos que usuários com gostos similares gostaram. A abordagem dos autores, entretanto, poderia ser expandida com o intuito de gerenciar objetos de aprendizagem virtuais. Desta forma dispositivos desprovidos de leitores de RFID poderiam ser utilizados com o sistema proposto, além disso poderiam ser utilizadas outras tecnologias para prover comunicação com o banco de dados que armazena os materiais

didáticos já que no modelo atual a comunicação é realizada somente através do Modelo de Referência para Compartilhamento de Conteúdo Objeto (SCORM).

O trabalho de Li et al., (2012) apresenta um sistema de recomendação demográfico e sensível ao contexto para recuperar arquivos criados pelo usuário sobre tópicos de aprendizado de uma linguagem. Tais arquivos são definidos pelos autores como “uma forma de conhecimento ou uma experiência de aprendizado adquirida em nosso dia a dia”. Além disso, o sistema também é usado para recomendar arquivos de outros usuários que têm a mesma língua materna e estão aprendendo a mesma língua. Contudo, a abordagem dos autores não considera outros interesses do usuário na sugestão de tais arquivos, por fim o sistema não recomenda outros tipos de objetos de aprendizagem como livros, slides de uma aula ou artigos científicos.

3.4 Considerações Finais

Neste capítulo foram apresentados os principais conceitos relacionados a sistemas de recomendação e a integração dos mesmos dando origem a área de sistemas de recomendação sensíveis ao contexto. O capítulo mostrou a definição adotada neste trabalho e definida por Ricci et al., (2011), logo após foram mostradas as tarefas que um usuário executa e que podem ser auxiliadas pelos sistemas de recomendação, tais tarefas foram definidas no trabalho de Herlocker et al., (2004).

Em seguida foram mostrados o principal modelo utilizado por sistemas de recomendação tradicionais, a filtragem colaborativa, e o modelo definido por Adomavicius et al., (2005) que deriva da filtragem colaborativa e diferencia-se por prover um modelo capaz de incorporar várias dimensões de informação e não mais apenas informações do usuário e do item a ser recomendado.

Este modelo proporcionou o surgimento de vários trabalhos relacionados a esta dissertação que são discutidos na subseção anterior. Entretanto nota-se que tais trabalhos poderiam ser estendidos através do enriquecimento dos modelos utilizados, já que em alguns casos informações importantes para o domínio, como informações sobre objetos de aprendizagem não são modelados. Outra característica apresentada pelos trabalhos discutidos é que nem sempre são utilizadas tecnologias abertas (ex. Web Services, SOAP, XML) dificultando assim a integração de novas fontes de dados às soluções.

No próximo capítulo será apresentada abordagem para recomendação de objetos de aprendizagem proposta neste trabalho, nele são apresentados a especificação, a interface, o modelo ontológico de contexto e a arquitetura da solução.

4 ABORDAGEM SENSÍVEL AO CONTEXTO PARA RECOMENDAÇÃO DE RECURSOS EM AMBIENTES PERVASIVOS

Este trabalho tem por objetivo prover uma solução ao problema da sobrecarga de recursos enfrentado pela comunidade de uma universidade. Mais especificamente, tenta-se prover uma solução à sobrecarga cognitiva que as pessoas enfrentam ao tentar selecionar um conjunto relevante de objetos de aprendizagem que sejam de seu interesse.

Uma exemplificação deste problema ocorre quando um estudante vai até a biblioteca para emprestar um livro que esteja disponível e seja de seu interesse, porém devido à grande quantidade de livros na biblioteca o estudante acaba demandando uma parte considerável de seu tempo para encontrar um livro que muitas vezes não satisfará as necessidades do mesmo, ao final do processo o estudante sente-se intelectualmente cansado por ler e analisar uma quantidade considerável de trechos de livros. Além de livros, este trabalho considera como objetos de aprendizagem qualquer entidade virtual ou física que pode ser utilizada para aprendizagem, educação e treinamento (“IEEE Standard for Learning Object Metadata,” 2002).

Para prover uma solução ao problema descrito foi criada uma abordagem sensível ao contexto e adaptativa para recomendação de objetos de aprendizagem à comunidade de uma universidade. Através desta abordagem espera-se diminuir o tempo dispendido por um estudante ou professor na busca de objetos de aprendizagem que melhor atendam suas necessidades, outrossim espera-se evitar a sobrecarga cognitiva que tal busca pode acarretar aos mesmos. A abordagem desenvolvida será melhor detalhada nas subseções deste capítulo, através da apresentação do modelo conceitual, da interface, do modelo ontológico de contexto desenvolvido e da arquitetura da solução.

4.1 Especificação da Solução

O sistema proposto foi desenvolvido para dispositivos que utilizam a plataforma *Android* e trabalha de maneira pervasiva estando integrado a vida do usuário sem que o mesmo perceba-o (WEISER, 1991), por causa desta característica ele é apto para recomendar objetos de aprendizagem em qualquer lugar que o usuário esteja ainda que tal recomendação não seja solicitada. Entretanto, devido a sua interface adaptativa o mesmo torna tais recomendações não intrusivas permitindo que o usuário selecione quais recomendações lhe interessam.

Além de tais características o sistema ainda é sensível ao contexto e por causa desta característica o mesmo é capaz de examinar e reagir à mudanças no ambiente (SCHILIT; ADAMS; WANT, 1994). O sistema reage a tais mudanças provendo recomendações de objetos de aprendizagem relevantes a atual situação do usuário.

A variável de contexto que é examinada durante o processo de recomendação é a localização do usuário e dos objetos de aprendizagem. Apesar de alguns objetos de aprendizagem existirem apenas de forma virtual eles têm sua localização associada a algum prédio dentro da universidade que estejam relacionados, por exemplo, as lâminas de uma aula são relacionadas ao prédio onde tal aula ocorre, um artigo científico está

relacionado a um prédio onde existe um laboratório ou uma sala de professor que pesquisem sobre o assunto, e assim por diante.

Por conta desta relação entre os objetos de aprendizagem e os prédios da universidade o sistema irá recomendar um objeto de aprendizagem quando um usuário estiver próximo ao prédio que ele está relacionado ou está presente fisicamente. Tais recomendações tendem a apresentar um número menor de objetos de aprendizagem quando comparados aos sistemas de recomendação tradicionais. Isto acontece porque nos sistemas tradicionais a recomendação é realizada levando-se em consideração apenas características do item a ser recomendado e do usuário.

Uma característica da abordagem aqui proposta é que ela leva em consideração o contexto do usuário ao realizar uma recomendação e isto faz com que a quantidade de objetos de aprendizagem recomendados diminua já que o contexto age como um filtro extra ao processo de recomendação. Portanto, ao invés de recomendar todos os objetos de aprendizagem significativos, ou seja aqueles que são do mesmo domínio de conhecimento do usuário, o sistema recomenda somente os objetos que são significativos e estão próximos à localização do usuário, evitando assim que o mesmo seja cognitivamente sobrecarregado.

Setten, Pokraev, Koolwaaij (2004) classificam o tipo de critério utilizado para selecionar os itens que serão recomendados ao usuário como rígido ou flexível. Um sistema adota um critério rígido quando o mesmo limita o conjunto de itens disponíveis; os itens que não correspondem ao critério rígido são descartados. Já o critério flexível é utilizado para ordenar o conjunto de itens selecionados ou para apresentar uma pontuação para o usuário em relação a cada item.

Por exemplo, a informação de localização pode ser utilizada como elemento de contexto para selecionar exclusivamente os itens dentro de uma certa distância do usuário (critério rígido); a informação de localização pode ser utilizada também para diminuir a relevância predita (pontuação) de um item de acordo com a distância deste ao usuário, quanto mais longe ele estiver menor será sua relevância (critério flexível).

A informação de contexto e os interesses do usuário são utilizados neste trabalho como um critério rígido para a seleção dos objetos de aprendizagem. Por conta disto, apenas os objetos que estiverem em prédios adjacentes ao prédio que o usuário está localizado e que correspondam aos interesses deste, serão recomendados. O critério flexível é bastante utilizado por sistemas de recomendação que implementam uma função de predição de relevância, como as demonstradas no Capítulo 3. Tais funções não são utilizadas por este trabalho pois o mesmo utiliza uma base de conhecimento e regras lógicas estabelecidas sobre os dados desta base para selecionar de maneira rígida os itens recomendados.

A base de conhecimento utilizada pela abordagem é composta do modelo ontológico de contexto, um banco de dados de informações contextuais que é alimentado com informações coletadas do ambiente e do usuário, um motor de inferências lógicas e regras descritas em forma de lógica de primeira ordem. Esta base de conhecimento tem como saída um conjunto de objetos de aprendizagem recomendados ao usuário, bem como a localização de tais objetos. Tais informações são apresentadas ao usuário através da aplicação desenvolvida.

4.1.1 Funcionalidades do sistema

As principais funcionalidades do sistema são listadas abaixo:

- O sistema deve perceber a localização do usuário. Tal informação é obtida através de leituras de dados do GPS disponível no dispositivo ou através do roteador wireless que o usuário está conectado ou caso tais opções falhem o sistema pede que o usuário informe sua posição manualmente.
- O sistema deve conhecer o perfil do usuário. As informações relativas ao perfil do usuário são coletada do banco de dados da universidade através de um webservice e aquelas informações que não estão disponíveis no banco de dados são informadas manualmente pelo usuário. Tais informações participarão do processo de recomendação onde será realizada uma correspondência entre os interesses do usuário e o domínio de conhecimento dos objetos de aprendizagem disponíveis.
- O sistema deve conhecer os objetos de aprendizagem. As informações relativas a tais objetos de aprendizagem são coletadas do banco de dados da plataforma de apoio a aprendizagem utilizada pelos cursos da universidade, do banco de dados das bibliotecas da universidade e de repositórios de objetos de aprendizagem compatíveis com a especificação IEEE LOM. A comunicação entre tais bancos de dados e o sistema é feita através de webservices. Tais objetos representam os itens alvo do processo de recomendação.
- O sistema deve exibir a localização atual do usuário e dos prédios próximos ao mesmo através de um mapa. O prédio mais próximo da localização do usuário e os prédios adjacentes a este deverão apresentar os objetos de aprendizagem recomendados.
- O sistema deve modificar, no mapa, as cores dos prédios relacionados aos objetos de aprendizagem recomendados com o intuito de destacá-los na interface apresentada ao usuário.
- O sistema deve listar, abaixo do mapa, os objetos de aprendizagem recomendados categorizados por tipo de objeto (ex. livro, lâminas de aulas, artigos científicos).
- O sistema deve agir de maneira pervasiva, recomendando objetos de aprendizagem ao usuário em qualquer lugar dentro da universidade mesmo que o usuário não solicite explicitamente tais recomendações.
- O sistema não deve ser intrusivo e deve permitir que o usuário selecione quais recomendações ele está interessado através de sua *interface*.

4.2 Interface da Solução

Na Figura 4.1 e na Figura 4.2 é apresentada a interface adaptativa do sistema, tal interface utiliza as informações providas pelas dimensões de contexto com o intuito de modificar a cor dos prédios próximos ao usuário. Ao alterar a cor dos prédios que contêm objetos de aprendizagem de interesse de um determinado usuário e que estejam próximos deste, o sistema produz o seu efeito de adaptação, ou seja, comporta-se diferente para diferentes usuários (BRUSILOVSKY; MILLÁN, 2007). Por conta desta característica dois usuários distintos podem passar próximos ao mesmo prédio e um deles receber recomendações de objetos de aprendizagem vinculados àquele prédio enquanto o outro não recebe recomendação alguma.

A interface proposta tem dois componentes principais: i) um mapa customizado, que mostra os prédios da universidade e a localização do usuário; ii) uma lista de objetos de aprendizagem recomendados. No primeiro componente, um marcador é utilizado para

mostrar a atual posição do usuário, além do marcador o prédio mais próximo da atual localização do mesmo tem sua cor alterada para vermelho. Os prédios adjacentes ao prédio da atual localização do usuário e que possuam objetos de aprendizagem recomendados vinculados têm sua cor alterada para amarelo.

Quando o usuário clica em um prédio amarelo, ou seja, aceita recomendação e deseja ver os objetos de aprendizagem de seu interesse que estão vinculados àquele prédio, o prédio tem então sua cor alterada de amarelo para azul. Outro marcador é apresentado, dentro do prédio azul, exibindo a localização do objeto de aprendizagem escolhido pelo usuário a partir de uma lista de opções. É importante notar que este marcador de objetos de aprendizagem só é exibido para objetos físicos, para o caso de objetos virtuais a modificação executada na interface é apenas colorir o prédio que o mesmo está relacionado de azul e apresentá-lo na lista de opções da mesma maneira feita para objetos físicos. Todos os outros prédios que não são relevantes atualmente são mantidos em sua cor preta original.

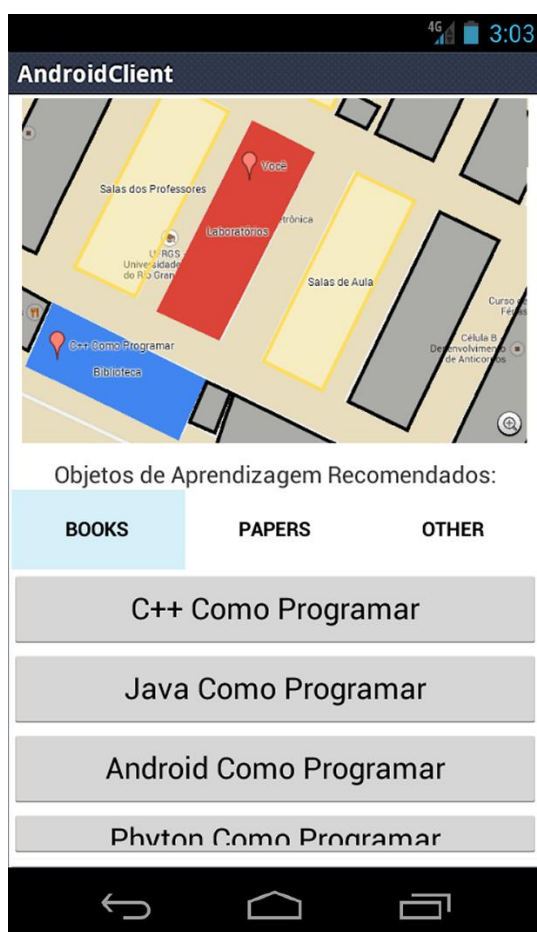


Figura 4.1: Interface exibindo o mapa e a lista de objetos de aprendizagem recomendados ao usuário

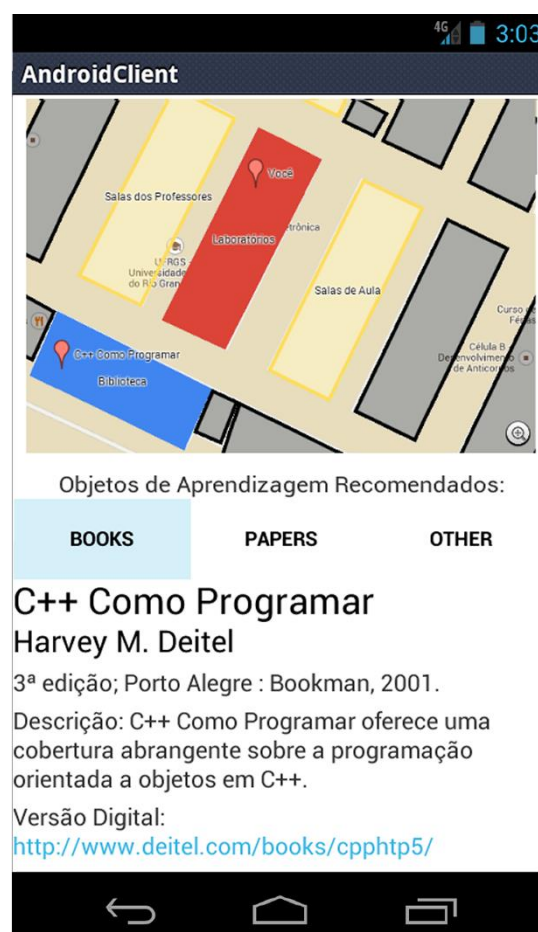


Figura 4.2: Interface exibindo o mapa e os detalhes do objeto de aprendizagem selecionado pelo usuário

O segundo componente é a lista de objetos de aprendizagem recomendados, apresentado na Figura 4.1, a lista é exibida quando um usuário clica em um dos prédios amarelos (que contém objetos de aprendizagem relacionados a ele) e o prédio tem sua cor alterada para azul. Sempre que uma lista de objetos recomendados aparecer na tela, ela

estará relacionada ao único prédio de cor azul apresentado pela interface. A lista apresenta os objetos recomendados categorizados através do seu tipo e divididos em abas.

Após o usuário selecionar um dos objetos de aprendizagem listados maiores detalhes de tal objeto são exibidos, como mostrado na Figura 4.2, e se o objeto existe apenas de forma digital é apresentado um link para o mesmo, se o objeto existe apenas de forma física é mostrada sua localização no mapa e se o objeto existe tanto de fisicamente quanto virtualmente as duas opções são exibidas na interface.

O exemplo exibido na Figura 4.1 mostra um usuário localizado no prédio dos laboratórios que aceitou verificar as recomendações de objetos de aprendizagem relacionados ao prédio da biblioteca. Assim que o usuário verifica a lista de livros recomendados e seleciona o livro chamado “C++ Como Programar” e a lista de objetos recomendados é substituída por informações adicionais do livro, como mostrado Figura 4.2 além de um marcador mostrar a localização aproximada do livro físico.

Apesar da possibilidade de estender a interface proposta um dos cuidados tomados durante sua construção foi mantê-la simples e intuitiva, tentando evitar que o usuário tenha que fazer um esforço adicional ao ter que memorizar o funcionamento da mesma. Para atingir tal objetivo Nielsen (1994) identificou dez heurísticas a serem seguidas com o objetivo de desenvolver interfaces de software dotadas de usabilidade. Essas heurísticas ficaram conhecidas pelo nome de 10 heurísticas de usabilidade de Nielsen, são elas:

- *Visibilidade do status do sistema:* o sistema deve sempre atualizar o usuário sobre o que está acontecendo através de feedbacks em um tempo razoável;
- *Correspondência entre o sistema e o mundo real:* o sistema deve falar a linguagem do usuário, ao invés de expor termos técnicos e orientados ao sistema. Siga convenções do mundo real apresentando as informações de forma natural e em uma ordem lógica;
- *Controle do usuário e liberdade:* usuários podem selecionar uma função por engano e por conta disso deve ser provida uma função de saída de emergência, para que o mesmo não precise passar por extensos diálogos em vão.
- *Consistência e padrões:* o usuário não deve ter que imaginar se palavras, ações ou situações diferentes significam a mesma coisa. Siga convenções da plataforma;
- *Prevenção de erros:* melhor que boas mensagens de erro é um *design* que previne que os erros aconteçam. Prevenir os erros é melhor também do que identificar situações onde possam acontecer erros e exibir mensagens de confirmação ao usuário antes dele executar tal ação;
- *Reconhecer ao invés de recordar:* minimize a sobrecarga da memória do usuário fazendo objetos, ações e opções visíveis. O usuário não deve ter que lembrar das coisas de uma seção para outra. Instruções de uso devem estar sempre visíveis e fáceis de recuperar quando necessário;
- *Flexibilidade e eficiência de uso:* acelere a interação, usuários novatos geralmente não veem isto, ações podem ser customizadas pelo usuário mais experiente, desta forma acelerando sua interação com o sistema;
- *Estética e design minimalista:* diálogos não devem ter informações irrelevantes ou raramente necessárias. Cada informação desnecessária compete com as informações realmente importantes e prejudicam sua visibilidade;

- *Ajude o usuário a reconhecer, diagnosticar e se recuperar de erros:* mensagens de erro devem utilizar linguagem natural, sem códigos, e quando possível sugerir uma solução;
- *Ajuda e documentação:* é melhor se o sistema puder ser utilizado sem documentação, entretanto qualquer ajuda deve ser facilmente encontrada, focada na tarefa do usuário passo a passo e nunca muito grande.

O autor aponta que tais sugestões foram chamadas de “heurísticas” pois são mais lembretes do que regras a serem seguidas. No desenvolvimento da interface apresentada tentou-se seguir a heurística “Correspondência entre o sistema e o mundo real” através da utilização de um mapa para mostrar a localização do usuário e dos objetos de aprendizagem recomendados, uma maneira mais natural do que simplesmente apresentar o valor de um ponto no espaço.

Outra heurística seguida foi “Controle do usuário e liberdade” através da implementação da função voltar, utilizando o botão provido pelo *smartphone*, em qualquer tela do sistema. Ainda tomou-se cuidado com a heurística “Reconhecer ao invés de recordar”, já que o mapa que mostra as localizações está disponível todo o tempo e o usuário não precisa recordar a localização do objeto de aprendizagem selecionado. Por fim tentou-se seguir a heurística “Estética e design minimalista” deixando o sistema simples com poucos botões e somente as informações realmente necessárias para apresentar o objeto de aprendizagem e sua localização.

4.3 Modelo Desenvolvido

O modelo de contexto desenvolvido neste trabalho encaixa-se na categoria de modelo aplicado, já que o mesmo tem por intenção ser utilizado em um ambiente pervasivo de ensino, no caso um campus universitário. Este modelo é utilizado por um sistema pervasivo que tenta solucionar o problema da dificuldade encontrada por uma pessoa na escolha de objetos de aprendizagem de interesse no ambiente da universidade.

O objetivo principal do modelo é descrever o contexto de um usuário de um sistema que recomenda objetos de aprendizagem de interesse. Os objetos recomendados podem ser físicos, como livros e apostilas, ou ainda virtuais, como slides e artigos acadêmicos.

O planejamento para construção do modelo baseou-se nos trabalhos relacionados discutidos na seção anterior. Como já falado notou-se um padrão de dimensões de informações presentes em modelos de contexto propostos para ambientes pervasivos, tais informações eram relativas ao perfil do usuário, ao ambiente e ao domínio tecnológico. Porém poucos trabalhos modelavam informações de objetos não computacionais envolvidos no contexto do usuário.

Tais objetos são de fundamental importância quando os modelos são utilizados para apoiar sistemas de recomendação, já que esses objetos representam os itens que serão recomendados ao usuário. Portanto, o modelo aqui proposto inova ao incluir informações de objetos não computacionais aos tradicionais modelos de contexto propostos na área de sistemas pervasivos e ao utilizar tal modelo ontológico para apoiar um sistema de recomendação.

Nas próximas subseções são apresentados a metodologia utilizada para construção do modelo, o modelo final desenvolvido e as ontologias reutilizadas para construção do mesmo. Posteriormente, no Capítulo 4, é descrita a integração do modelo ao sistema de recomendação desenvolvido.

4.3.1 Metodologia utilizada

A metodologia utilizada para construir o modelo ontológico apresentado neste trabalho foi proposta por Noy; McGuinness (2001) e foi nomeada de “*Ontology development 101*”. Os autores propõem uma maneira simples e iterativa para construção de uma ontologia. Porém, deve ser frisado que não existe uma maneira correta de modelar um domínio e sim alternativas viáveis, além disso a construção de ontologias é um processo iterativo e por fim, conceitos na ontologia devem assemelhar-se a objetos (físicos ou virtuais) e relacionamentos do domínio de interesse. A metodologia é dividida em sete passos principais a serem seguidos durante a construção do modelo ontológico.

- Passo 1 – Determine o domínio e o escopo da ontologia: para definir o escopo e o domínio da ontologia os autores sugerem que se responda i) Qual é o domínio que a ontologia irá cobrir? ii) Para que a ontologia será utilizada? iii) Para quais tipos de questões a informação presente na ontologia deve prover respostas? iv) Quem irá utilizar e manter a ontologia? Os autores ainda sugerem que devam ser realizadas questões mais específicas das quais a ontologia desenvolvida deverá ser capaz de responder. Alguns exemplos de questões específicas para uma ontologia de vinhos são: O vinho bordô é branco ou vermelho? O buquê ou o corpo de um vinho mudam dependendo do ano de colheita?
- Passo 2 – Considere reutilizar ontologias: quase sempre vale a pena considerar o que outras pessoas já fizeram e checar se é possível estender ou refinar outras ontologias para o domínio em questão. Este passo pode se tornar um requisito caso o sistema a ser desenvolvido necessite interagir com outros sistemas que já estão comprometidos com outras ontologias. Outra vantagem alcançada através da reutilização é a possibilidade de tornar o modelo mais compartilhável, já que outros sistemas que se baseiam no modelo reutilizado podem ser mais facilmente integrados ao sistema desenvolvido. Utilizar um sistema de representação de conhecimento pode ajudar nesta tarefa, já que muitos desses sistemas permitem a manipulação de várias ontologias em um mesmo ambiente independente do formalismo em que a ontologia é expressa.
- Passo 3 – Enumere termos importantes na ontologia: é importante fazer uma lista de termos importantes do domínio, estes termos geralmente aparecem em sentenças ou devem ser explicados ao usuário. Os autores sugerem algumas que podem ajudar a definir tais termos, são elas: Sobre quais termos gostaríamos de falar? Quais propriedades têm esses termos? O que gostaríamos de falar sobre esses termos? Não é necessário se preocupar com questões de sobreposição de conceitos que tais termos representam, relações entre os termos, ou propriedades que tal conceito deve ter, ou ainda se determinado conceito refere-se a uma classe ou a uma propriedade durante a formulação dos termos neste passo. Os próximos dois passos são geralmente executados em paralelo e de maneira iterativa já que estão fortemente relacionados. O que geralmente acontece é que se defina um conjunto pequeno de classes e logo em seguida defina-se suas propriedades, posteriormente são acrescentadas outro conjunto pequeno de classes e definidas suas propriedades e assim por diante.

- Passo 4 - Defina as classes e a hierarquia de classes: existem várias abordagens para se desenvolver uma hierarquia de classes i) *top-down*, primeiro são definidos os conceitos mais genéricos do domínio e posteriormente são definidos os conceitos mais específicos; ii) *bottom-up*, primeiro são definidos os conceitos mais específicos do domínio e posteriormente os mesmos são agrupados em conceitos mais genéricos; iii) híbrida, é uma combinação das duas técnicas anteriores, onde são definidos os conceitos mais importantes do domínio e posteriormente os mesmos são especializados ou generalizados conforme o caso. Para iniciar a definição das classes do domínio toma-se a lista de termos criada no passo 3 e são identificados os termos que definem objetos de existência independente ao invés de termos que descrevem características desses objetos. Tais termos serão então definidos como classes. Uma classe A será uma superclasse de B se uma instância de B puder ser também considerada uma instância de A.
- Passo 5 – Defina propriedades (slots) das classes: somente a definição das classes não é suficiente para responder as questões elaboradas no passo 1, deve ser definida a estrutura interna dos conceitos. Após a seleção de termos que se tornarão classes, executada no passo 3, selecionam-se os termos restantes para descrição de propriedades (ex. cor, sabor, aroma) dessas classes. Tais propriedades de classes podem: i) ser de natureza intrínseca ou extrínseca ao conceito a que são relativas; ii) descrever partes físicas ou abstratas do conceito; iii) representar relacionamentos dos indivíduos pertencentes a uma determinada classe com indivíduos de outra. Uma propriedade deve estar relacionada com a classe mais genérica que possua tal propriedade, pois assim todas as classes derivadas de tal classe genérica herdarão tal propriedade.
- Passo 6 – Defina as características/restrições dos slots: uma propriedade (slot) de uma classe pode ter ainda várias características, algumas das mais comuns são i) cardinalidade do slot, define quantos valores um slot pode ter, em alguns sistemas é possível determinar que um slot pode ter um ou múltiplos valores enquanto em outros é possível determinar um número mínimo e um número máximo de valores para determinado slot; ii) tipo de valor do slot, descreve com quais tipos de valores um slot pode ser preenchido (ex. *string*, número, booleano, enumeração, tipo instância); iii) domínio e alcance do slot, um slot ter um tipo de valor instância significa que o indivíduo daquela classe se relacionará com indivíduos das classes determinadas pelo tipo instância, tais classes determinadas pelo tipo instância são chamadas de “alcance” do slot, já as classes que o slot está relacionado são chamadas de domínio do slot;
- Passo 7 – Crie instâncias: este é o último passo para a criação de uma ontologia, nele são criadas instâncias individuais de uma classe, para isso deve-se escolher uma classe, criar a instância individual da classe escolhida e preencher os valores dos slots (propriedades) relativas à instância.

Um dos passos de fundamental importância na elaboração do modelo proposto neste trabalho é o passo 2 (Considere reutilizar ontologias). Por conta disso na próxima subseção será apresentado o modelo desenvolvido e na subseção posterior serão apresentadas as ontologias reutilizadas durante a construção do modelo.

4.3.2 Estrutura do modelo

Conforme explicado na seção anterior o modelo proposto neste trabalho tem por objetivo apoiar um sistema de recomendação de objetos de aprendizagem em uma universidade. Sistemas de recomendação tradicionais utilizam pelo menos informações sobre o usuário e o item a ser recomendado para gerar suas recomendações (RICCI; ROKACH; SHAPIRA, 2011). Já os sistemas pervasivos, conforme mostrado nos trabalhos relacionados deste capítulo, utilizam pelo menos informações do usuário, do ambiente e dos recursos computacionais para modelar o contexto.

O modelo ontológico de contexto aqui proposto foi construído em OWL, através do software *Protégé*¹, e tem o usuário como seu elemento central. O modelo é apresentado em quatro dimensões de informação contextuais i) informações sobre o perfil do usuário; ii) informações sobre localização; iii) informações sobre os elementos tecnológicos e iv) informações sobre os objetos de aprendizagem.

O modelo de contexto é apresentado na Figura 4.3 e mostra a estrutura básica de classes da ontologia desenvolvida, porém algumas dessas foram suprimidas da imagem com o intuito de deixá-la menos poluída e facilitar seu entendimento. Note na figura, que o usuário tem um perfil, tem uma localização de interesse e usa uma tecnologia e devido a tais fatores ele tem objeto de aprendizagem recomendado, fato exibido através da relação de propriedade objeto “temOAREcomendado”.

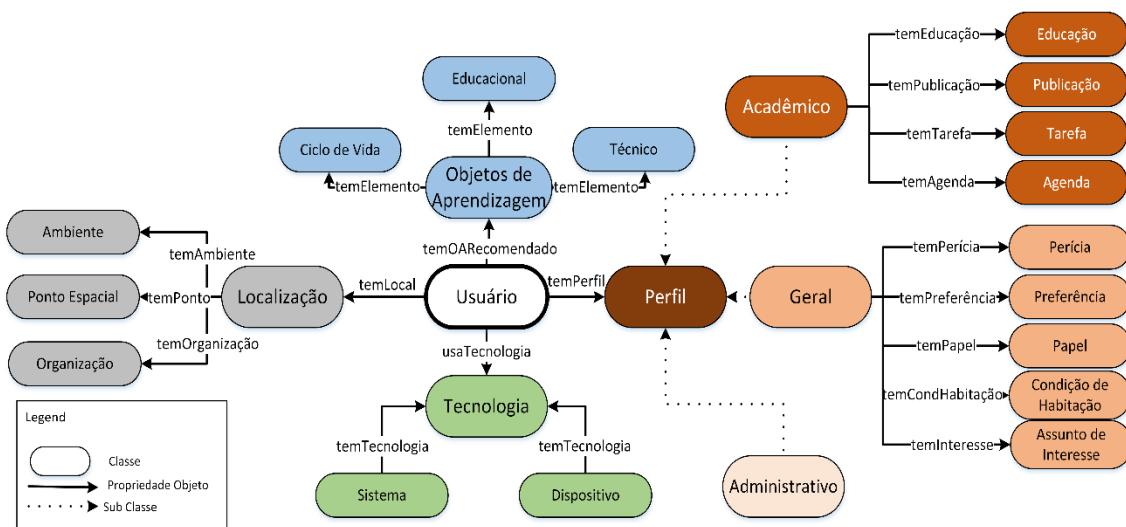


Figura 4.3: Visão sumarizada do modelo de contexto proposto com suas quatro dimensões de informação

As dimensões de informações contextuais são exibidas em maiores detalhes a partir da Figura 4.4. Além da estrutura de classes as figuras mostram também os principais relacionamentos do tipo propriedade objeto e todos os relacionamentos de subclasse.

Devido à complexidade em definir as informações pertinentes ao perfil do usuário o trabalho de Brusilovsky e Millán (2007) propõe cinco características populares e úteis em modelos de usuários de sistemas adaptativos, são elas: i) o conhecimento do usuário, que desempenha papel fundamental em sistemas educacionais adaptativos; ii) os interesses do usuário, é a parte mais importante do perfil do usuário para sistemas de recomendação

¹ <http://protege.stanford.edu/>

web; iii) os objetivos e tarefas, representam o propósito do usuário em um sistema adaptativo, dependendo do tipo de sistema podem representar o objetivo do trabalho, uma necessidade de informação imediata ou um objetivo de aprendizagem; iv) o plano de fundo do usuário, é um conjunto de características que foge aos conceitos centrais do modelo, por exemplo a profissão do usuário, responsabilidades no trabalho e experiência em áreas relacionadas, este tipo de informação é geralmente utilizada para adaptar o conteúdo de um software; e v) os traços individuais, são responsáveis por definir um usuário individualmente, representam características intrínsecas tais como, traços de personalidade, estilos cognitivos, e estilos de aprendizagem.

Apesar do trabalho de Brusilovsky e Millán (2007) ser focado na definição de características importantes a perfis de usuários de sistemas adaptativos, fica claro que tais características podem ser utilizadas para a modelagem de perfis de usuário de sistemas de recomendação.

As peculiaridades do domínio aqui estudado e do problema a que este trabalho se propõe a resolver (recomendar objetos de aprendizagem em universidades), sugerem que as características de interesse e plano de fundo do usuário podem ser utilizadas no alcance de uma solução eficaz. Por conta disso a dimensão de informações contextuais de perfil do usuário aqui apresentada será focada nessas duas características, apesar de apresentar também conceitos de outras características (ex. tarefas e agenda).

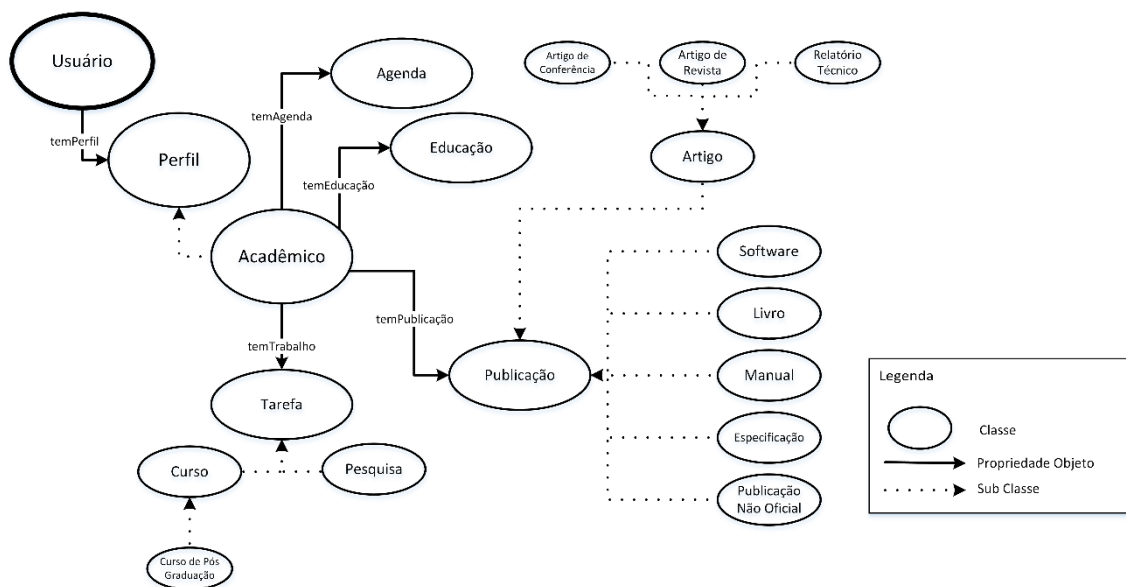


Figura 4.4: Sub-dimensão contextual de informações do perfil acadêmico do usuário

A dimensão de informações contextuais de perfil do usuário concentra o maior número de conceitos e relacionamentos entre as dimensões do modelo, com o intuito de organizar esses conceitos em categorias esta dimensão foi dividida em sub-dimensão de perfil acadêmico, sub-dimensão de perfil administrativo e sub-dimensão de perfil geral.

Na Figura 4.4 é exibida a sub-dimensão de perfil acadêmico que é composta pelos seguintes conceitos:

- “Agenda”, responsável por descrever a agenda com o horário de aulas e demais compromissos do usuário;

- “Educação”, responsável por descrever o título de grau do usuário (ex. Dr., MSc);
- “Publicação”, responsável por descrever a produção científica do usuário. Este conceito é pai dos conceitos “Artigo”, “Software”, “Livro”, “Manual”, “Especificação” e “Publicação Não Oficial”.
- “Tarefa”, responsável por descrever o trabalho realizado pelo usuário dentro da universidade. Este conceito está relacionado com conceito “Agenda” no sentido que ele é responsável por descrever os itens que serão marcados na agenda.

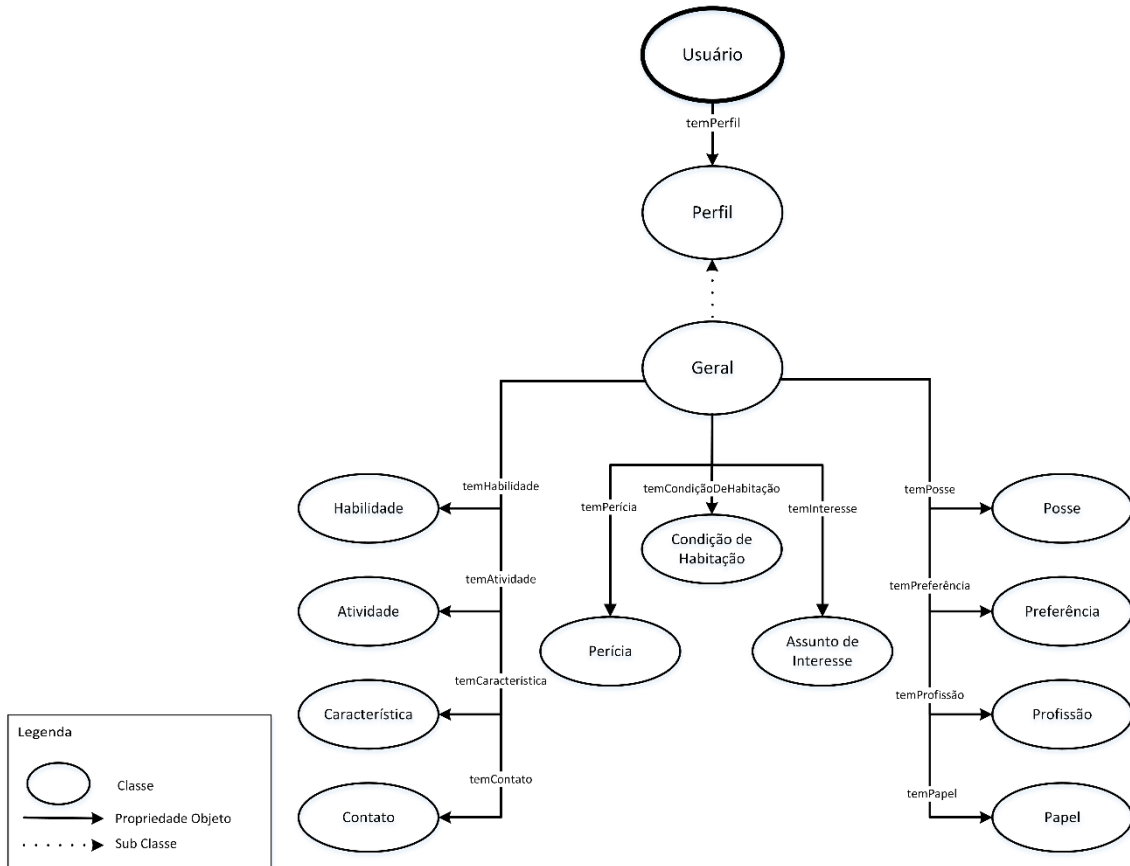


Figura 4.5: Sub-dimensão contextual de informações do perfil geral do usuário

A sub-dimensão de informações contextuais do perfil geral do usuário é exibida na Figura 4.5 e é composta pelos seguintes conceitos:

- “Habilidade”, descreve as habilidades e incapacidades do usuário, tanto mentais quanto físicas;
- “Atividade”, descreve as atividades, os passatempos e os trabalhos relacionados do usuário (ex. colecionar selos, investigar sobre a segunda guerra);
- “Característica”, descreve características gerais do usuário (ex. cor dos olhos, altura, peso)
- “Contato”, descreve outras pessoas que o usuário se relaciona, tais como, família, amigos e colegas de trabalho;
- “Perícia”, descreve conhecimentos que o usuário possui e que demonstra habilidade, destreza e maestria (ex. o usuário é perito em computadores);

- “Condição de Habitação”, descreve informações importantes do local de residência e do tipo de casa do usuário;
- “Assunto de Interesse”, descreve interesses de passatempos ou de assuntos acadêmicos que o usuário possa ter. Além de descrever interesses do usuário esse conceito também se relaciona através de uma propriedade objeto ao conceito que descreve os objetos de aprendizagem, isso porque tais objetos pertencem a um domínio do conhecimento que pode ser relativo a um assunto de interesse do usuário;
- “Posse”, descreve as coisas vivas e não vivas que o usuário possa ter ou ser relacionado (ex. carro, livro, gato);
- “Preferência”, descreve os gostos do usuário (ex. gosta de música erudita, não gosta de funk);
- “Profissão”, descreve a ocupação profissional do usuário.
- “Papel”, descreve o papel do usuário dentro da universidade (ex. diretor do instituto de informática, reitor, professor)

A sub-dimensão de informações contextuais do perfil administrativo do usuário aparece no modelo desenvolvido como um único conceito a ser estendido caso haja necessidade. Esta sub-dimensão descreve informações que variam de acordo com a instituição de ensino pois são altamente dependentes das características da secretaria e da administração de cada instituição. Algumas sugestões de informações relacionadas a esta dimensão são pendências de documentos do aluno, informações bancárias dos bolsistas, cota de impressões do aluno e etc.

As informações ambientais, ou de localização, constituem a segunda dimensão contextual do modelo. Esta dimensão é tão importante que Brusilovsky e Millán (2007) afirmam que os sistemas sensíveis ao contexto para dispositivos móveis focam-se naturalmente na localização do usuário. Diferentemente das outras informações que dão base para a adaptação do conteúdo, da apresentação ou do estilo navegacional (BRUSILOVSKY; MAYBURY, 2002), a informação de localização é utilizada para determinar um conjunto pequeno de objetos de interesse próximo a localização do usuário. Este subconjunto define o que deve ser apresentado ou recomendado ao usuário.

A Figura 4.6 exibe a dimensão contextual de informações do ambiente, esta dimensão é constituída dos seguintes conceitos:

- “Ambiente”, descreve os ambientes/áreas dentro da universidade (ex. sala de aula, auditório, laboratório). Se relaciona com conceito “Tipo de Ambiente” através da propriedade objeto “temTipoDeAmbiente”;
- “Tipo de Ambiente”, descreve um ambiente como público ou privado;
- “Organização”, descreve o papel desempenhado por um conjunto de ambientes da universidade (ex. faculdade, departamento, instituto). Para descrever tais papéis o conceito se relaciona com o conceito “Ambiente” através de uma propriedade objeto “temAmbiente”;

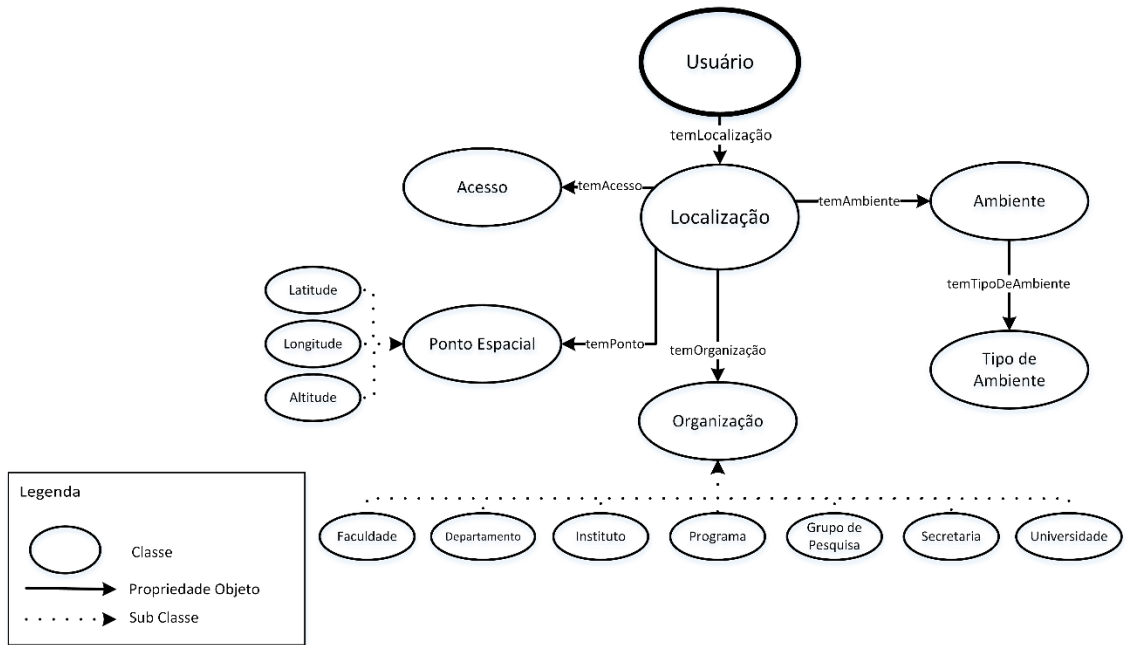


Figura 4.6: Dimensão contextual de informações do ambiente (localização)

- “Ponto Espacial”, descreve informações de GPS relativas a localização do usuário e do objeto de aprendizagem para isso utiliza os conceitos “Latitude”, “Longitude” e “Altitude”;
- “Acesso”, descreve o tipo de acesso ao ambiente como permitido ou negado;

A dimensão de localização se relaciona com o conceito “Ambiente”, “Usuário” e a dimensão de objetos de aprendizagem através da propriedade objeto “temLocalização”.

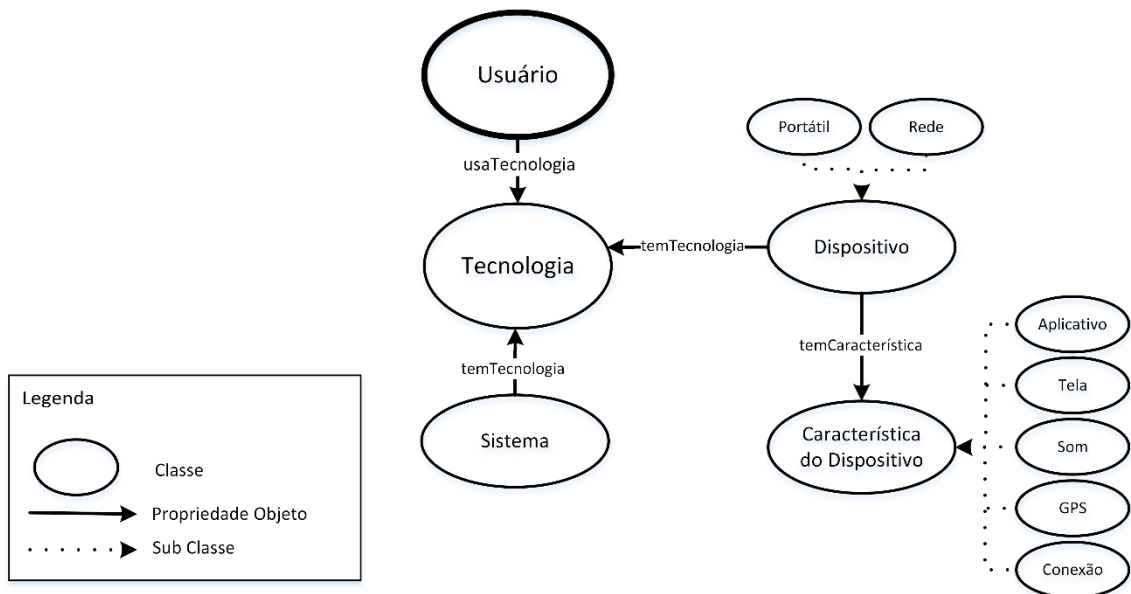


Figura 4.7: Dimensão contextual de informações dos recursos computacionais

A dimensão contextual de informações dos recursos computacionais é responsável por descrever os recursos de tecnologia presentes no ambiente pervasivo, bem como os sistemas presentes neste ambiente. Dotada de um número consideravelmente menor se

comparada às dimensões anteriores, esta dimensão é exibida na Figura 4.7 e é composta pelos seguintes conceitos:

- “Sistema”, descreve o próprio sistema de recomendação e os demais sistemas presentes no ambiente, apresenta informações como nome do sistema e versão;
- “Dispositivo”, descreve o dispositivo utilizado para acessar o sistema de recomendação, bem como todos os dispositivos espalhados pela universidade (ex. computadores, sensores, atuadores, roteadores). É pai dos conceitos “Portátil” (descreve dispositivos portáteis) e “Rede” (descreve dispositivos de rede). Apresenta um relacionamento “temSistema” do tipo propriedade objeto com o conceito “Sistema”;
- “Característica do Dispositivo”, descreve características funcionais do dispositivo (ex. tamanho da tela, tipos de arquivos suportados). É pai dos conceitos “Aplicativo” (descreve os aplicativos suportados pelo dispositivo), “Tela” (descreve características relativas a tela do dispositivo, como tamanho e resolução), “Som” (descreve os perfis sonoros do dispositivo, como silencioso, alto, normal), “GPS” (descreve o modelo do GPS utilizado pelo dispositivo) e “Conexão” (descreve todos os tipos de conexão que o dispositivo é capaz de estabelecer, como *bluetooth*, *wireless*, *ethernet*, RFID).

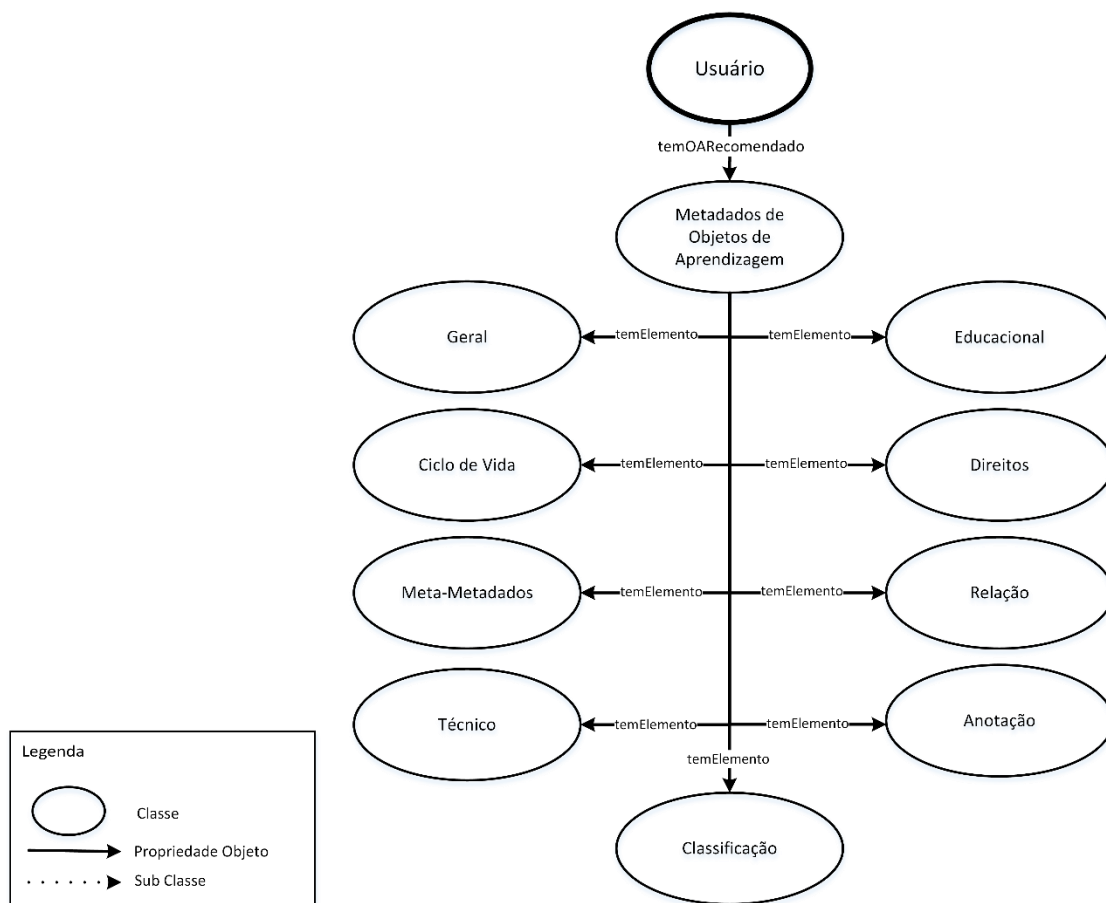


Figura 4.8: Dimensão contextual de informações dos objetos de aprendizagem

Por fim tem-se a dimensão contextual de informações dos objetos de aprendizagem, exibida na Figura 4.8, esta dimensão descreve as características de um objeto de aprendizagem através de nove conceitos, são eles:

- “Geral”, descreve informações gerais que caracterizam o objeto de aprendizagem como um todo;
- “Ciclo de Vida”, descreve características relativas ao histórico e ao estado corrente do objeto de aprendizagem atual, além disso, descreve as mesmas características dos objetos de aprendizagem que afetaram o objeto atual durante sua evolução;
- “Meta-Metadados”, descreve informações sobre os outros conceitos da dimensão;
- “Técnico”, descreve requisitos e características técnicas de objetos de aprendizagem;
- “Classificação”, descreve o objeto de aprendizagem atual em relação a uma determinada classificação;
- “Anotação”, descreve comentários de uso educacional a respeito do objeto de aprendizagem, além de descrever por quem e quando o comentário foi criado;
- “Relação”, descreve características que relacionam um objeto de aprendizagem a outro;
- “Direitos”, descreve direitos de propriedade intelectual e condições de utilização do objeto de aprendizagem;
- “Educativa”, descreve características educacionais e pedagógicas do objeto de aprendizagem;

4.3.3 Modelos reutilizados

Foram reutilizados dois modelos ontológicos e uma especificação de metadados para a construção do modelo de contexto apresentado anteriormente, tais modelos reutilizados são apresentados em maiores detalhes a partir da Figura 4.9. A reutilização de tais modelos foi possível devido a função importação de ontologias e de conversão de arquivos *Resource Description Framework* (RDF) em OWL do software *Protégé*, já que o modelo final foi desenvolvido totalmente em OWL, com o intuito de possibilitar a execução de inferências lógicas sobre o conhecimento descrito no mesmo.

Um dos modelos reutilizados foi o padrão de metadados para objetos de aprendizagem definidos pela IEEE (“*IEEE Standard for Learning Object Metadata*,” 2002). Apesar de ser encontrado apenas em forma de um documento PDF o padrão é muito bem descrito, o que tornou possível a construção do mesmo em OWL.

O objetivo do padrão é facilitar a busca, avaliação e aquisição de objetos de aprendizagem por aprendizes, instrutores ou ferramentas automatizadas de software. Além de cumprir tais objetivos, o padrão ainda habilita o compartilhamento e troca de objetos de aprendizagem através do desenvolvimento de catálogos e inventários de tais objetos.

Na Figura 4.9 é exibido a estrutura hierárquica dos conceitos modelados na ontologia a partir do padrão. Estes conceitos formam a dimensão contextual de informações dos objetos de aprendizagem. Devido ao nível de maturidade do padrão o mesmo pôde ser utilizado em sua totalidade e sem a necessidade qualquer adaptação de seu conteúdo durante o desenvolvimento do modelo de contexto deste trabalho.

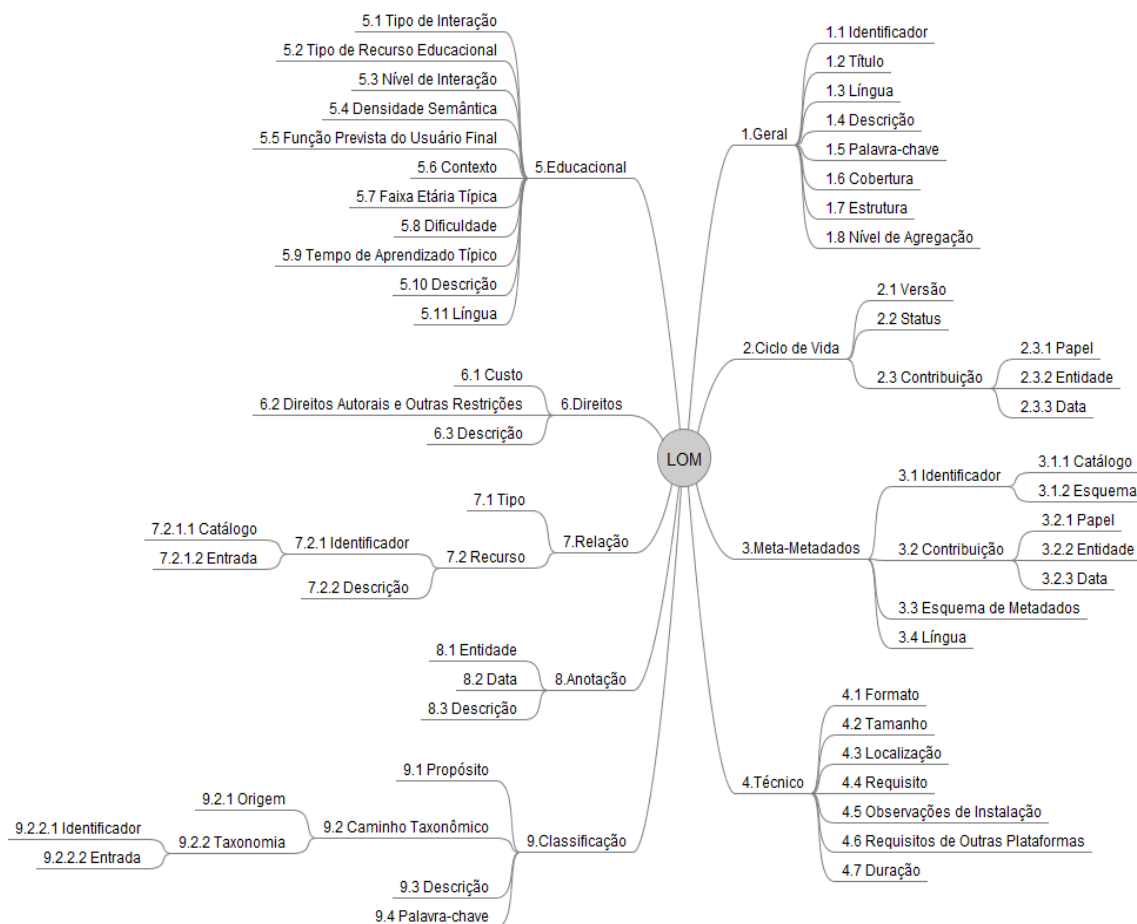


Figura 4.9: Estrutura hierárquica dos conceitos descritos no modelo IEEE LOM

Outro modelo reutilizado por este trabalho foi o modelo desenvolvido por Guo, Pan, Heflin (2005), batizado de *Lehigh University Benchmark* (LUBM), encontrado em OWL, o modelo tem por objetivo ser utilizado para avaliar o desempenho e escalabilidade de sistemas baseados em conhecimento. Para alcançar tal objetivo os autores desenvolveram um modelo ontológico de uma universidade genérica. Apesar de ter sido desenvolvido para testes de desempenho de motores de inferência o modelo apresentado na Figura 4.10 mostra-se interessante principalmente para a modelagem da estrutura hierárquica de uma universidade.

Foram necessárias algumas modificações para utilização deste modelo, a primeira decisão tomada foi separar seus conceitos e juntá-los à dimensão de informação que mais se adequasse. O conceito “Pessoa” foi renomeado como “Papel” e ele junto a seus conceitos filhos foram integrados à sub- dimensão contextual de informações de perfil geral. O conceito “Trabalho” foi renomeado como “Tarefa” e ele junto a seus conceitos filhos foram integrados à sub-dimensão contextual de informações de perfil acadêmico. Os conceitos “Agenda” e “Publicação” junto a seus conceitos filhos foram integrados à sub-dimensão contextual de informações de perfil acadêmico. Por fim o conceito “Organização” e seus conceitos filhos foram integrados à dimensão contextual de informações do ambiente.

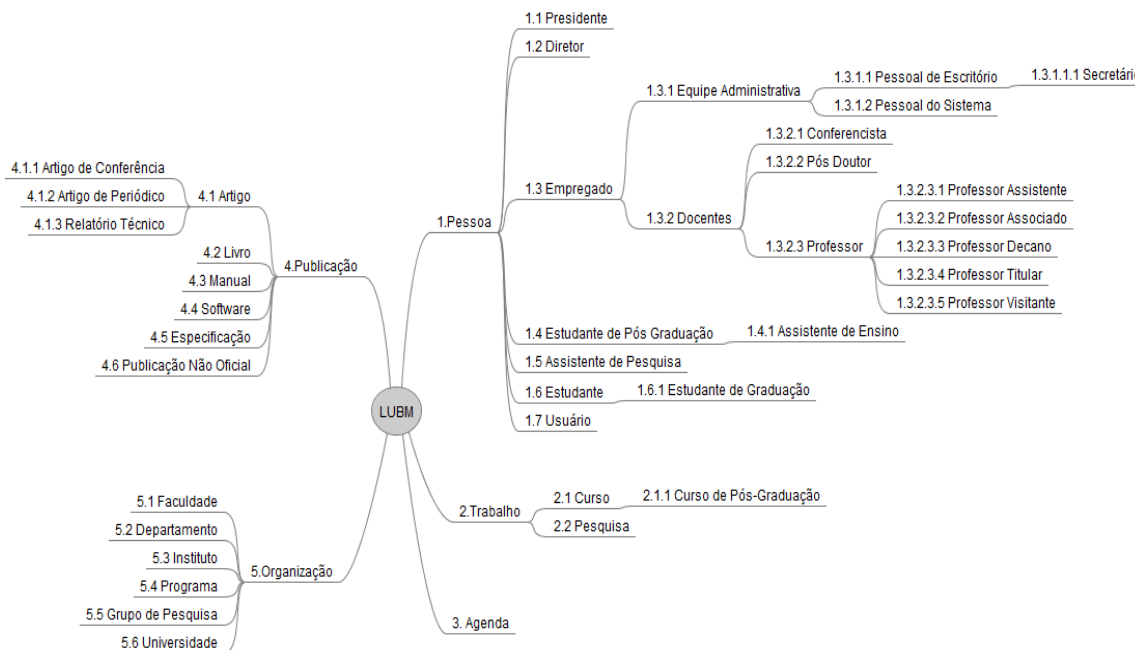


Figura 4.10: Estrutura hierárquica dos conceitos descritos na ontologia LUBM de Guo, Pan, Heflin (2005)

O terceiro modelo reutilizado neste trabalho foi desenvolvido por Golemati et al. (2007) e tem por objetivo descrever de maneira geral, em um modelo ontológico, conceitos e propriedades utilizadas para modelar o perfil de um usuário. O modelo original foi desenvolvido em RDF e foi convertido para OWL através da ferramenta *Protégé*.

A Figura 4.11 mostra a estrutura hierárquica dos conceitos deste modelo. Um dos conceitos mais importantes encontrados neste modelo é o conceito “Interesse”, este conceito é utilizado durante o processo de recomendação, melhor descrito no Capítulo 4, e teve seu nome alterado para “Assunto de Interesse” possibilitando assim a criação de um relacionamento do tipo propriedade objeto entre ele e o conceito “Metadados de Objetos de Aprendizagem”. Os demais conceitos do modelo foram alocados na sub-dimensão contextual de informações de perfil geral, com exceção do conceito “Educação” e seus conceitos filhos que foram alocados na sub-dimensão contextual de informações de perfil acadêmico.

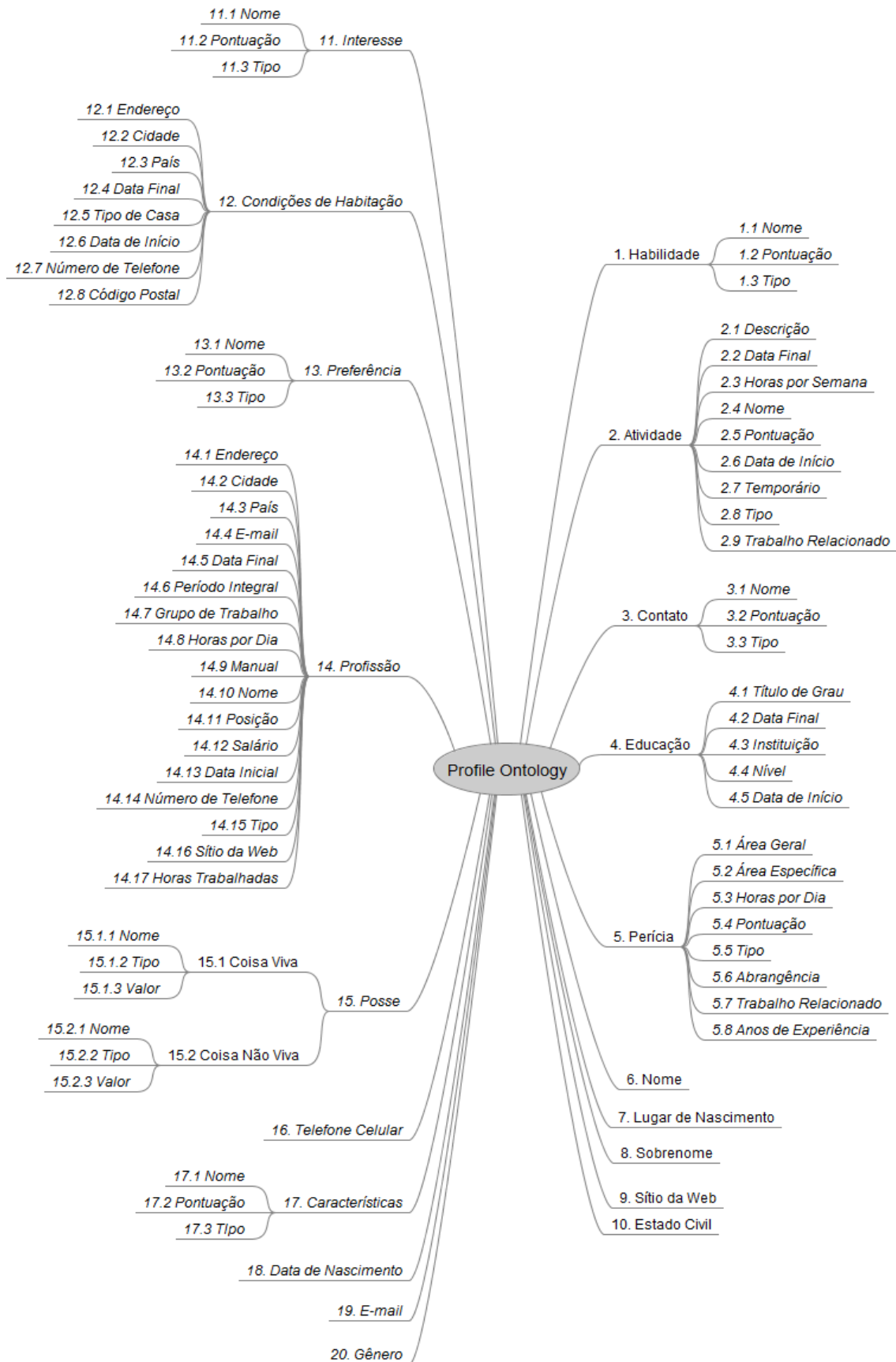


Figura 4.11: Estrutura hierárquica dos conceitos descritos na ontologia de perfil do usuário de Golemati et al. (2007)

4.4 Arquitetura da Solução

A arquitetura da solução proposta neste trabalho leva em consideração um modelo cliente-servidor e é dividida em três componentes, como exibido na Figura 4.12, o Coletor de Contexto e o Recomendador Baseado em Conhecimento (Recomendador BC) no lado do servidor e o Controlador de Interface no lado do cliente.

A comunicação entre os componentes do cliente e do servidor é feita através da utilização de *web services*, que são definidos por Ferris (2004) como sistemas de software projetados para apoiar a interação entre máquinas tornando-as interoperáveis em uma rede. Uma das vantagens ao utilizar a tecnologia de *web services* é tornar os componentes do lado servidor aptos a integrar-se a outro cliente, permitindo que no futuro a solução esteja disponível para dispositivos de diferentes plataformas operacionais.

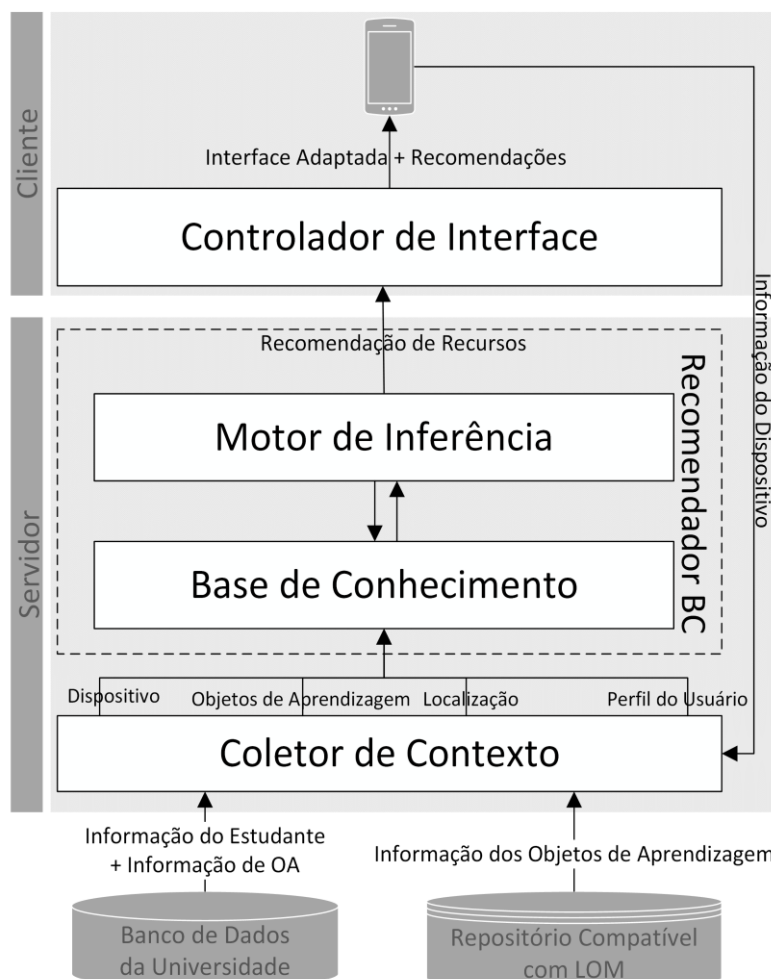


Figura 4.12: Arquitetura da Solução

A comunicação entre os componentes é iniciada assim que o usuário abre o sistema e começa a interagir com a interface adaptativa. Quando o usuário está se deslocando de um prédio em direção a outro, de maneira que ele esteja mais próximo ao prédio de destino do que o prédio de origem, o sistema então envia ao Coletor de Contexto os dados atuais do dispositivo e da localização do usuário. A informação de localização é obtida através das leituras do GPS ou através do roteador *wireless*, quando porém essas opções não estão disponíveis o sistema não consegue notificar o Coletor de Contexto sobre a mudança significativa da localização do usuário, por conta disto o mesmo deve informar

manualmente sua atual localização para dar início ao processo de recomendação. Assim que recebe tais dados o coletor recupera informações do usuário e dos objetos de aprendizagem disponíveis através do banco de dados da universidade, o componente recupera ainda informações de objetos de aprendizagem disponíveis em repositórios de software que são compatíveis com o padrão de metadados de objetos de aprendizagem LOM.

Uma vez que o Coletor de Contexto finalizou a recuperação das informações ele as repassa ao Recomendador Baseado em Conhecimento (Recomendador BC), mais especificamente ao modelo ontológico de contexto armazenado neste componente que irá criar instâncias dos conceitos relativos às informações recebidas.

Após instanciar o modelo, o Motor de Inferência será acionado para raciocinar sobre as regras lógicas, descritas em *Semantic Web Rule Language* (SWRL), presentes na Base de Conhecimento. SWRL é uma linguagem definida em Horrocks et al. (2004) que tem por objetivo estender a sintaxe da OWL ao incluir regras de lógica de primeira ordem aumentando a capacidade de representação semântica da OWL. As regras SWRL do modelo realizam a correspondência entre os interesses do usuário e o domínio de conhecimento do objeto de aprendizagem a ser recomendado.

O resultado produzido pelo Motor de Inferência é um conjunto de pares (usuário, objeto de aprendizagem recomendado) que é enviado por *web services* ao Controlador de Interface junto com tal informação ainda é enviada a localização de cada objeto de aprendizagem recomendado.

Por fim o Controlador de Interface, organiza as informações recebidas criando uma lista de objetos de aprendizagem recomendados e vinculando cada objeto da lista a uma localização no mapa. A interface é então exibida ao usuário, que interage com a mesma e reinicia o processo de recomendação e adaptação. Cada componente da arquitetura é vinculado às tecnologias que o apoiam e é melhor detalhado abaixo:

- *Coletor de Contexto*: é o componente responsável por recuperar as informações que serão utilizadas durante o processo de recomendação, tais informações são relacionadas às quatro dimensões definidas pelo modelo de contexto. As informações são recuperadas através da comunicação com o dispositivo do usuário e com bancos de dados e repositórios externos, toda a comunicação ocorre através do uso de *web services*. A utilização de tal tecnologia permite a introdução de novas repositórios de objetos de aprendizagem e bancos de dados, além de permitir que clientes de diferentes plataformas sejam desenvolvidos e anexados ao servidor especificado. Este componente é construído em Java e utiliza o servidor *Apache Tomcat*.
- *Recomendador BC*: é formado pela Base de Conhecimento e o Motor de Inferência. A Base de Conhecimento é composta pelo modelo ontológico de contexto, as instâncias do modelo e as regras de recomendação SWRL. Este componente recebe as informações de contexto e é responsável pela criação das instâncias do modelo. Para instanciar o modelo ontológico de contexto foi utilizada a *OWL API*. O Motor de Inferência é responsável por raciocinar sobre as regras de recomendação SWRL presentes na Base de Conhecimento e produzir o conjunto de pares (usuário, objeto de aprendizagem). O motor de inferência utilizado para raciocinar sobre as regras foi o *Pellet*.
- *Controlador de Interface*: é o único componente presente no lado do cliente. Este componente recebe os pares gerados pelo motor de inferência junto com

a localização dos objetos de aprendizagem e organiza tais informações na interface da aplicação que é então apresentada ao usuário. Este componente é construído em Java e utiliza o *Android Development Kit (SDK)* e a *Google Maps API*.

A arquitetura é construída no modelo cliente-servidor com o intuito de evitar o *overhead* da instanciação do modelo ontológico e da execução do motor de inferência em um dispositivo móvel com memória e poder de processamento limitados.

4.5 Considerações Finais

Uma universidade é um ambiente cheio de recursos físicos e virtuais, a quantidade desses recursos disponíveis entretanto pode levar uma pessoa a sentir-se cognitivamente sobrecarregada durante a escolha de alguns deles. Por causa desta sobrecarga a pessoa acaba escolhendo um conjunto de recursos que não irá atender suas necessidades e preferências da melhor maneira. Objetos de aprendizagem são recursos importantes dentro de uma universidade e a escolha desses pode também levar a uma sobrecarga cognitiva.

Um sistema de recomendação surge neste cenário como uma solução para auxiliar a escolha de tais objetos de aprendizagem de maneira a evitar a sobrecarga cognitiva enfrentada pelas pessoas durante a seleção de tais objetos. A utilização de ontologias e de informações de contexto mostram-se eficazes na recomendação de recursos em ambientes pervasivos. Através da utilização de dispositivos é possível coletar informações de localização do usuário e sugerir objetos de aprendizagem relacionados aos prédios próximos. Um objeto de aprendizagem está relacionado a um prédio através de sua presença física no mesmo ou através de outros fatores, como uma aula que acontece em um prédio relaciona as lâminas daquela aula àquele prédio.

A abordagem proposta neste trabalho tem por objetivo recomendar objetos de aprendizagem que estejam relacionados a prédios próximos a atual localização do usuário e que sejam de interesse deste. Para realizar tal recomendação são utilizadas informações sobre os interesses do usuário, as características do item e a localização atual do usuário, tais informações estão modeladas em uma ontologia de contexto.

A ontologia de contexto é instanciada através de informações coletadas em bancos de dados externos ao sistema e dados obtidos do dispositivo móvel do usuário. Uma vez instanciada a ontologia é verificada por um motor de inferência que raciocina sobre regras, escritas em SWRL, que fazem uma correspondência entre os interesses do usuário e o domínio de conhecimento do objeto de aprendizagem.

O resultado obtido pelo motor de inferência é um conjunto de pares (usuário, objeto de aprendizagem recomendado), tal conjunto é unido à localização dos objetos recomendados e enviado a um controlador de interface que organiza as informações e as apresenta ao usuário através de uma interface adaptativa.

A interface adaptativa exibe um mapa da localização atual do usuário e dos prédios próximos a ele que são relacionados à objetos de aprendizagem recomendados. Os prédios que contêm objetos recomendados e o prédio mais próximo da localização atual do usuário têm suas cores alteradas no mapa. Assim que o usuário aceita verificar quais objetos recomendados estão relacionados a um determinado prédio é exibida uma lista de objetos de aprendizagem que quando selecionados exibem detalhes sobre os mesmos.

No próximo capítulo é apresentado o estudo de caso desenvolvido para validar o modelo e a abordagem proposta. No capítulo é apresentado um cenário de aplicação, uma comparação entre motores de inferência disponíveis, uma regra de recomendação genérica e uma aplicada ao cenário.

5 ESTUDO DE CASO

Neste capítulo é desenvolvido um estudo de caso que tem por objetivo validar o modelo de contexto e a abordagem propostos. Em Thomas (2011) é encontrada uma definição genérica de estudo de caso que adaptada à realidade deste trabalho define que um estudo de caso é uma análise do processo de recomendação feita através de um método holístico, ou seja, que tenta compreender o processo de recomendação como um todo e não apenas de maneira isolada.

Para a realização do estudo de caso apresenta-se um cenário de uso onde é descrito um caso que demonstra como uma recomendação é feita e as consequências que tal recomendação exerce no ambiente acadêmico. O cenário de uso trata de uma instância do processo, onde um estudante recebe recomendações de objetos de aprendizagem dentro da universidade.

Logo após a descrição do cenário de uso é apresentado um estudo comparativo entre os principais motores de inferência compatíveis com a ferramenta *Protégé*. Por fim é apresentado o filtro semântico sensível ao contexto, construído a partir de regras SWRL que é responsável pela execução das recomendações e a aplicação deste filtro no cenário descrito.

5.1 Cenário de Uso

Apesar da abordagem proposta poder ser utilizada em vários cenários de recomendação de objetos de aprendizagem, dentro do ambiente universitário, foi selecionada uma situação específica para validação do modelo e apresentação da regra de recomendação relacionada. Neste cenário William é um estudante do curso de Ciência da Computação e está matriculado na disciplina de “Introdução a Linguagens de Programação Orientadas a Objetos”. O professor responsável pela disciplina então atribuiu como tarefa que os alunos buscassem e implementassem um algoritmo de ordenação de elementos de um vetor utilizando a linguagem de programação Java.

Após o término da aula William atualizou seus interesses no sistema incluindo a linguagem de programação Java como um deles. Mais tarde, o sistema percebeu a localização do usuário e verificou, através da correspondência de interesses do usuário e do domínio de conhecimentos dos objetos de aprendizagem próximos, que o livro “Java Como Programar” poderia ser recomendado e estava relacionado a um prédio próximo a atual localização de William. O prédio relacionado ao livro, no caso o prédio da biblioteca central, tem então sua cor alterada na interface do sistema. William checa as recomendações que o sistema apresenta e clica sobre o prédio da biblioteca. Por fim a lista de objetos de aprendizagem recomendados é apresentada ao usuário que acaba escolhendo obter mais detalhes do livro de Java.

No cenário apresentado a recomendação de um livro interessante e disponível pode levar o estudante ao prédio da biblioteca para aluga-lo e assim completar a tarefa atribuída pelo professor. Tal recomendação colabora no sentido de evitar uma sobrecarga cognitiva do usuário porque serão recomendados apenas objetos de aprendizagem específicos para a situação atual, ou seja, próximo ao prédio da biblioteca. Em um sistema de recomendação tradicional seriam recomendados todos os objetos de aprendizagem de interesse do usuário, ainda que tais objetos não fossem de interesse no contexto atual.

Uma generalização do cenário apresentado acima é mostrada através da Figura 5.1, nela é exibido o ambiente da universidade e o usuário, que está passando próximo a um dos prédios da universidade. O sistema cliente recebe informações a respeito de sua localização atual através do GPS e envia tais informações ao servidor que junta a informação de localização às informações do usuário e dos objetos de aprendizagem, instancia o modelo de contexto e raciocina sobre as regras SWRL e devolve um conjunto de objetos de aprendizagem com suas respectivas localizações ao cliente. O usuário então recebe tais recomendações enquanto ainda está próximo ao prédio relacionado aos objetos de aprendizagem recomendados. Por fim, o usuário seleciona os objetos que deseja obter mais informações.

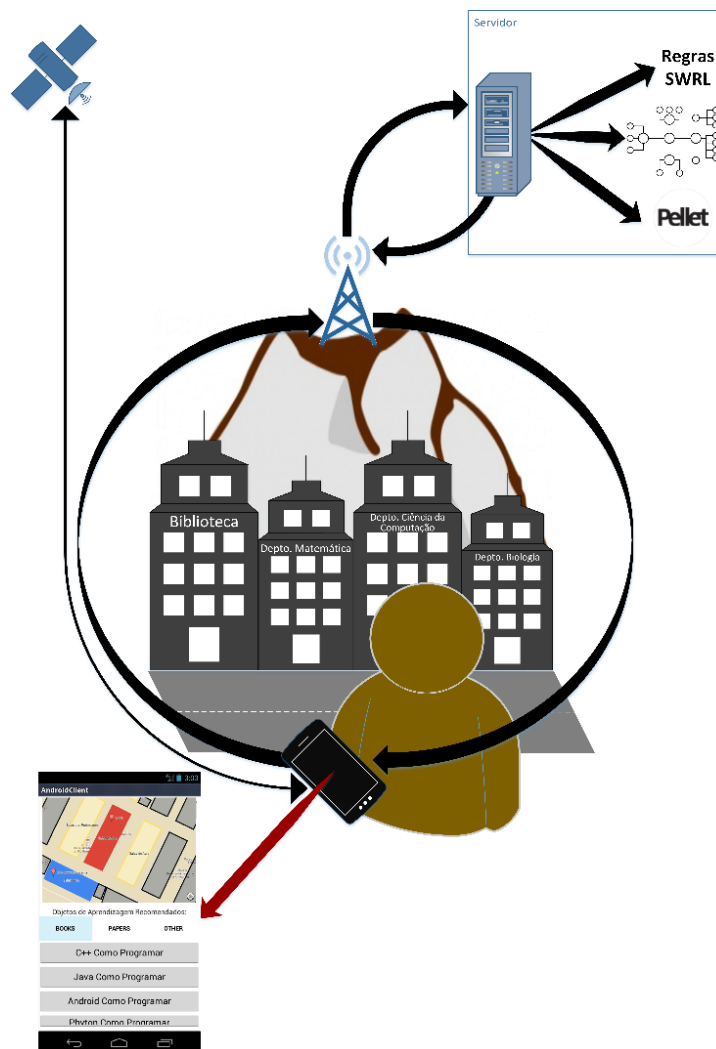


Figura 5.1: Visão geral do funcionamento da solução

Os objetos de aprendizagem recomendados são apresentados ao usuário através da interface não intrusiva do sistema, isto é, eles estão relacionados aos prédios próximos e serão exibidos apenas se o usuário aceitar a recomendação, clicando em tal prédio. Portanto, além do filtro provido pela informação de contexto, que irá apresentar apenas os objetos próximos ao usuário, o usuário aplica um último filtro nas recomendações selecionando apenas os prédios que deseja ver objetos relacionados.

5.2 Motores de Inferência

Os motores de inferência são ferramentas de fundamental importância pois são utilizados em conjunto com as regras de recomendação escritas em SWRL e tornam possível a filtragem dos objetos de aprendizagem próximos ao usuário. Um motor de inferência é um programa que infere consequências lógicas a partir de um conjunto explícito de fatos afirmados ou axiomas e tipicamente provê suporte automatizado para tarefas de raciocínio tais como classificação, depuração e consulta (DENTLER et al., 2011).

Tabela 5.1: Notações de linguagem descritiva

Notação	Descrição
S	ALC estendido com papéis transitivos
H	Hierarquia de papéis
O	Nominais (permite restrições de valor de objetos – owl:oneOf, owl:hasValue)
Q	Restrição de cardinalidade qualificada (restringe o de valores de um determinado tipo que é relacionado a um papel – “tem exatamente quatro partes que são pernas”)
N	Restrição de cardinalidade (restringe o número total de valores para um papel - owl:cardinality, owl:maxCardinality)
(D)	Uso de propriedades de tipo de dados
A	Conceito atômico
AL	Variedade da lógica descritiva, significa linguagem atributiva (permite: negação atômica, interseção de conceitos, restrições universais, quantificação existencial limitada)
U	União de conceitos
C	Negação complexa de conceitos
E	Qualificação existencial completa
F	Propriedades funcionais
EL	Variedade da lógica descritiva (permite: interseção de conceitos e restrições existenciais de quantificação existencial completa)
I	Construtor de papéis inversos
EL+	Uma variedade menos expressiva da lógica descritiva que pode ser resolvida em tempo polinomial
H	Hierarquia de papéis com herança simples
R	Inclusão limitada de axiomas de papéis complexos; reflexividade e irreflexibilidade; papel disjunção.

Fonte: Adaptado de ABBURU, 2012. p. 34.

Motores de inferência utilizam lógica descritiva para inferir consequências lógicas. A lógica descritiva é uma família de linguagens formais para representação de conhecimento, tal lógica modela conceitos (classes), papéis (propriedades) e indivíduos (instâncias). O conceito de modelagem fundamental da lógica descritiva é o axioma, que é uma afirmação lógica que relaciona papéis e/ou conceitos (BAADER, 2003). A lógica descritiva provê os construtores de conceito Booleanos que quando combinados aos construtores de restrição existencial e universal são chamados de ALC, onde os construtores de conceitos Booleanos são o conceito de disjunção, o conceito de conjunção e o conceito de negação. A lógica descritiva ALC corresponde ao fragmento da lógica de

primeira ordem obtido ao restringir a sintaxe à fórmulas contendo duas variáveis. Outros construtores de lógica descritiva são exibidos na Tabela 5.1.

Os motores de inferência podem ser classificados segundo uma série de características, tais como algoritmo de inferência, suporte lógico, grau de completude do raciocínio, linguagem de implementação, entre outros. Na Tabela 5.2 é exibida uma comparação entre nove dos mais comumente utilizados motores de inferência compatíveis com a ferramenta *Protégé*. Os atributos utilizados para fazer tal comparação são explicados em maiores detalhes a seguir:

- *Solidez e Completude*: Esta característica do motor de inferência avalia se todas as possíveis inferências são inferidas ou não. Estas características podem ajudar significativamente a acelerar o processo de raciocínio.
- *Expressividade*: Avalia se são permitidos diferentes tipos de axiomas de lógica descritiva, tais como axiomas de transitividade e axiomas de inclusão de papéis.
- *Perfil nativo*: indica fragmentos lógicos que provêm poder de expressividade para a eficiência do motor de inferência.
- *Classificação Incremental*: quando uma ontologia está sendo classificada e um axioma é atualizado (através de adições ou remoções) o motor de inferência utiliza uma classificação feita anteriormente e adiciona os axiomas modificados para produzir a nova hierarquia de conceitos.
- *Suporte a Regras*: possibilita que o motor de inferência combine ontologias com regras.
- *Plataformas*: indica com quais sistemas operacionais um motor de inferência pode funcionar.
- *Justificativas*: esta característica indica se um motor de inferência provê justificativas para inconsistências encontradas nas ontologias.
- *Raciocínio ABox*: indica raciocínio com indivíduos e inclui checagem de instâncias, resposta à consulta conjuntiva e checagem de consistência *ABox*.
- *OWL API*: é uma API para trabalhar com ontologias escritas em OWL. Provê uma interface padrão para motores de inferência, desta forma uma aplicação pode ter vários motores de inferência sem a necessidade de alterar sua implementação.
- *OWL Link*: permite tornar motores de inferência compatíveis com OWL API em servidores que podem ser acessados por aplicações compatíveis com OWL API.
- *Suporte ao Protégé*: indica se o motor de inferência pode ser utilizado com a ferramenta *Protégé* ou não.
- *Suporte ao NeOn*: indica se o motor de inferência pode ou não ser utilizado com a ferramenta de edição de ontologias *NeOn*.
- *Licença*: A maior diferença entre as licenças é se elas podem ser reconhecidas como *Open Source* ou não.
- *Suporte ao Jena*: indica se o motor de inferência pode ser utilizado com a API do *Jena* ou não.
- *Linguagem de implementação*: indica em qual linguagem o motor de inferência foi implementado.
- *Disponibilidade*: indica se o motor de inferência é disponibilizado gratuitamente ou não.

Tabela 5.2: Comparação de motores de inferência (SIM representa característica suportada, NÃO representa característica não suportada, S/N representa necessidade de melhor explicação)

		Pellet	RACER	FACT++	Snorocket	SWRL-IQ
Solidez		SIM	SIM	SIM	SIM	SIM
Compleitude		SIM	SIM	SIM	SIM	NÃO
Expressividade		SROIQ(D)	SHIQ	SROIQ(D)	EL+	-
Perfil Nativo		DL, EL	DL	DL	EL	-
Classificação Incremental	Adições	SIM	NÃO	NÃO	SIM	S/N
	Remoções	SIM	NÃO	NÃO	NÃO	S/N
Suporte a Regras		SIM (SWRL)	SIM (SWRL)	NÃO	NÃO	SIM (SWRL)
Plataformas		TODAS	TODAS	TODAS	TODAS	TODAS
Justificativas		SIM	SIM	NÃO	NÃO	SIM
Raciocínio ABox		SIM	SIM	SIM	NÃO	SIM
OWL API		SIM	SIM	SIM	SIM	NÃO
OWL Link API		SIM	SIM	SIM	NÃO	NÃO
Suporte ao Protégé		SIM	SIM	SIM	SIM	SIM
Suporte ao NeOn		SIM	NÃO	NÃO	NÃO	NÃO
Licença		DULI: AGPL	PRÓPRIA	GLGPL	PRÓPRIA	S/N
Suporte ao Jena		SIM	NÃO	NÃO	NÃO	NÃO
Linguagem de Implementação		JAVA	LISP	C++	JAVA	PROLOG
Disponibilidade		OPEN SOURCE	COMERCIAL	OPEN SOURCE	COMERCIAL	S/N

		HermiT	CEL	TrOWL	ELK	
Solidez		SIM	SIM	SIM	SIM	
Compleitude		SIM	SIM	SIM	SIM	
Expressividade		SROIQ(D)	EL+	SROIQ	EL	
Perfil Nativo		DL	EL	DL, EL	EL	
Classificação Incremental	Adições	NÃO	SIM	NÃO	SIM	
	Remoções	NÃO	NÃO	NÃO	SIM	
Suporte a Regras		SIM (SWRL)	NÃO	NÃO	SIM (FORMATO PRÓPRIO)	
Plataformas		TODAS	LINUX	TODAS	TODAS	
Justificativas		NÃO	SIM	NÃO	NÃO	
Raciocínio ABox		SIM	SIM	SIM	NÃO	
OWL API		SIM	SIM	SIM	SIM	
OWL Link API		SIM	SIM	NÃO	S/N	
Suporte ao Protégé		SIM	SIM	SIM	SIM	
Suporte ao NeOn		SIM	NÃO	NÃO	NÃO	
Licença		GLGPL	LICENÇA APACHE 2.0	DULI: AGPL	LICENÇA APACHE 2.0	
Suporte ao Jena		NÃO	NÃO	SIM	S/N	
Linguagem de Implementação		JAVA	LISP	JAVA	JAVA	
Disponibilidade		OPEN SOURCE	OPEN SOURCE	COMERCIAL	OPEN SOURCE	

Fonte: Adaptado de ABBURU, 2012. p. 37.

De acordo com os dados apresentados na Tabela 5.2 o motor de inferência Pellet destaca-se entre as outras ferramentas por ser o motor que possui o conjunto de características mais completo. Pellet é a única ferramenta, dentre as analisadas que possui a maior expressividade de lógica descritiva combinada com dois perfis nativos, total classificação incremental, suporte a regras SWRL, justificativas e compatibilidade com as três APIs verificadas, além de ser *Open Source*. Por prover o conjunto mais completo de características Pellet é o motor de inferência utilizado neste trabalho.

5.3 Regras de Recomendação

As regras de recomendação escritas em SWRL são responsáveis pela filtragem dos objetos de aprendizagem próximos ao usuário. Tais regras são escritas em forma de implicações lógicas entre um antecedente (corpo) e um conseqüente (cabeça) (HORROCKS et al., 2004). Um exemplo simples de regra SWRL é exibido a seguir:

$$pai(?x, ?y) \wedge irmão(?y, ?z) \rightarrow tio(?x, ?z)$$

A leitura da regra acima é feita da seguinte forma, se um indivíduo representado através da variável $?y$ é pai do indivíduo representado pela variável $?x$ e o indivíduo representado pela variável $?z$ é irmão de $?y$, então $?z$ é tio de $?x$.

Tais regras executam a correspondência entre os interesses do usuário e o domínio de conhecimento a que determinado objeto de aprendizagem pertence. Uma generalização do tipo de regra proposto neste trabalho é a seguinte:

$$\begin{aligned} &usuário(?u) \\ &\wedge temLocalização(?l, ?u) \\ &\wedge característicaASerCorrespondida(?c) \\ &\wedge temCaracterística(?c, ?u) \\ &\wedge objetoDeAprendizagem(?o) \\ &\wedge temLocalização(?x, ?o) \\ &\wedge temCaracterística(?c, ?o) \\ &\rightarrow temOAREcomendado(?o, ?u) \end{aligned}$$

Através dessa regra é verificado se o usuário $?u$ está na localização $?l$, ainda é verificado se o usuário possui uma característica $?c$, tal que $?c$ esteja modelada no perfil de $?u$ e possa ser correspondida com um domínio de aprendizagem, exemplos desse tipo de características encontradas no modelo de perfil aqui proposto são os interesses, habilidades, perícia e preferências do usuário. Logo após verificar a característica de perfil do usuário é verificado se o objeto de aprendizagem a ser recomendado $?o$ está na localização $?x$, note que o objeto e o usuário não necessariamente estarão no mesmo ponto espacial, já que a solução recomenda objetos próximos a atual localização do usuário. Por fim é verificado se o objeto de aprendizagem a ser recomendado $?o$ possui um domínio de aprendizagem que corresponda aos interesses, preferências, habilidades ou perícia do usuário.

Uma instanciação desta regra pode ser feita considerando-se como característica a ser correspondida o conceito “Assunto de Interesse”. O usuário se relaciona com este conceito através de seu perfil geral e os objetos de aprendizagem se relacionam com este mesmo conceito através de uma propriedade objeto, já que eles descrevem um assunto de interesse do usuário. A regra instanciada é exibida a seguir:

$$\begin{aligned}
& \text{usuário}(? u) \\
& \wedge \text{temLocalização}(\text{PertoBiblioteca}, ? u) \\
& \wedge \text{assuntoDeInteresse}(? c) \\
& \wedge \text{temAssuntoDeInteresse}(? c, ? u) \\
& \wedge \text{objetoDeAprendizagem}(? o) \\
& \wedge \text{temLocalização}(\text{Biblioteca}, ? o) \\
& \wedge \text{temAssuntoDeInteresse}(? c, ? o) \\
& \rightarrow \text{temOARecomendado}(? o, ? u)
\end{aligned}$$

A regra instanciada agora apresenta o relacionamento do tipo propriedade objeto “assuntoDeInteresse” como característica a ser correspondida entre o objeto de aprendizagem e o usuário. Outra particularidade da regra é a utilização de instâncias específicas do conceito localização ao invés de utilizar variáveis, isto é feito com o intuito de garantir através das regras que as localizações do usuário e do objeto de aprendizagem sejam aproximadas. Ao utilizar instâncias de localização, ao invés de variáveis, diretamente nas regras evita-se que o motor de inferências faça combinações entre duas localizações que não sejam próximas.

É possível utilizar as instâncias de localização direto nas regras porque as informações relativas à localização (ambientes, pontos espaciais, etc.) dentro da universidade são conhecidas e não variam com frequência, assim tais informações podem ser instanciadas no modelo e utilizadas para a construção das regras. Uma vantagem ao utilizar a informação instanciada direto na regra é a diminuição do espaço de busca de localizações, fazendo que a regra seja inferida com maior eficiência.

Na Figura 5.2 é exibida a ontologia geral instanciada para o cenário de uso apresentado anteriormente, note que em um primeiro momento William se relaciona com três dimensões de informação (Perfil, Localização e Tecnologia). Entretanto, William ainda não se relaciona com a dimensão de objetos de aprendizagem, já que este relacionamento será criado apenas após a execução da regra SWRL de recomendação. Na mesma figura ainda são mostrados em destaque os relacionamentos e conceitos relativos ao objeto de aprendizagem e sua localização, do usuário e sua localização além do assunto de interesse do usuário e do objeto de aprendizagem. Esses conceitos e relacionamentos estão destacados pois são os elementos da ontologia que são verificados pelo motor de inferência durante a execução da regra SWRL instanciada.

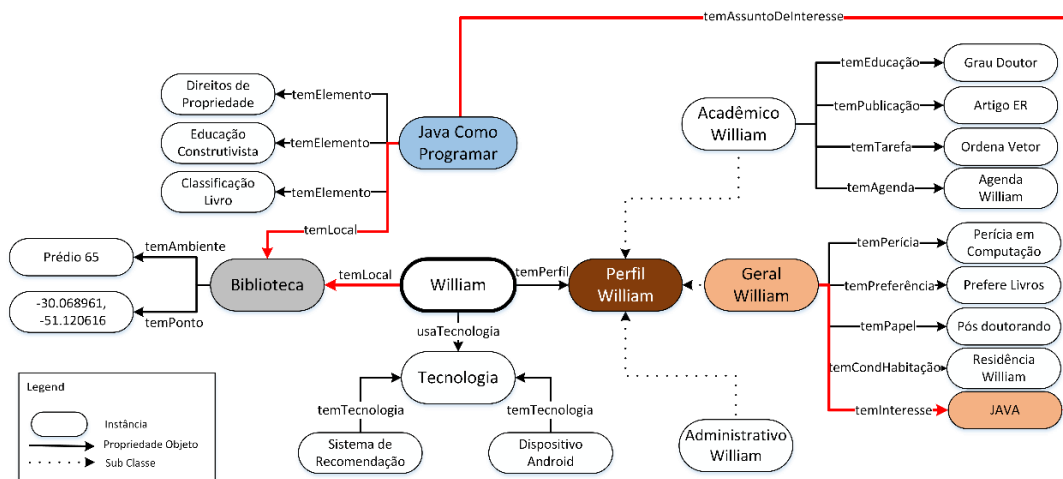


Figura 5.2: Ontologia geral instanciada segundo o cenário descrito

Após a execução da regra é então criado o relacionamento entre o usuário William e o objeto de aprendizagem “Java Como Programar” tal como descrito no cenário, este relacionamento é apresentado no modelo ontológico através da propriedade objeto “temOARecomendado”, como é mostrado em destaque na Figura 5.3.

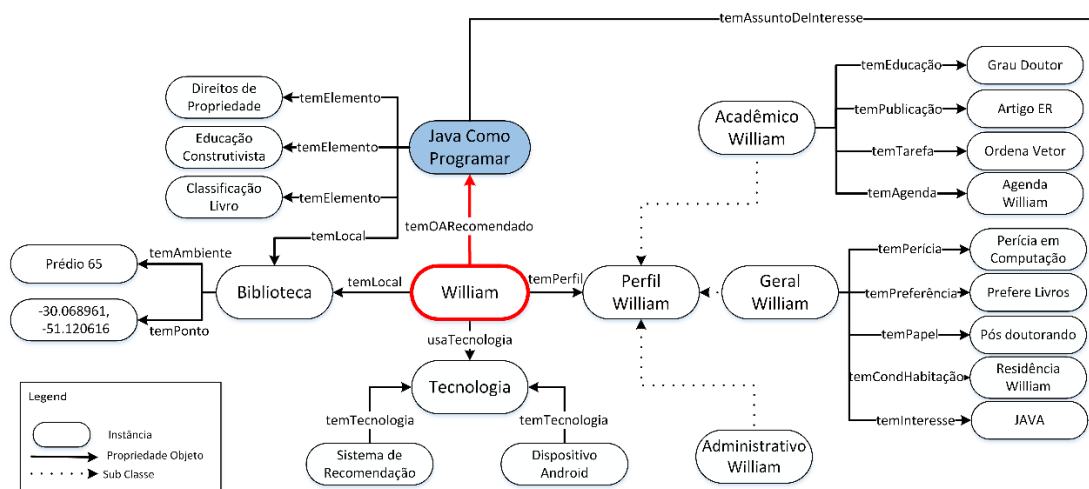


Figura 5.3: Ontologia instanciada após a execução da regra de recomendação SWRL

A Figura 5.4 exibe o resultado obtido ao utilizar o motor de inferência Pellet para raciocinar sobre a regra instanciada. Verifica-se que o motor de inferência conseguiu, através da utilização da regra e das instâncias do modelo, chegar ao resultado que o aluno William receberá, ao passar próximo à biblioteca, a recomendação do livro “Java Como Programar”. Como dito anteriormente, esta recomendação baseia-se na correspondência entre o assunto de interesse do usuário e do objeto de aprendizagem, além da localização aproximada dos dois, definida explicitamente na regra.

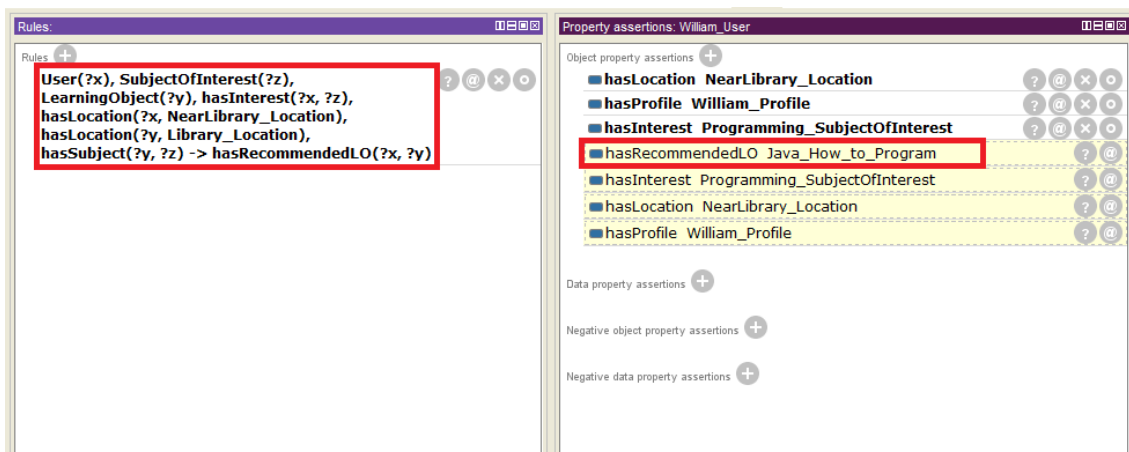


Figura 5.4: Resultado gerado pelo motor de inferência Pellet ao raciocinar sobre a regra SWRL instanciada

Ao utilizar a linguagem SWRL e a ontologia de contexto como artefatos para realizar a recomendação dos objetos de aprendizagem próximos ao usuário torna-se explícita toda a semântica do processo de recomendação. Por conta disto, toda a semântica da aplicação que estaria restrita ao código fonte é disponibilizada no modelo ontológico. Uma vantagem é que isto torna possível que o conhecimento do domínio e o comportamento

do sistema sejam entendidos tanto por agentes humanos quanto por agentes computacionais.

5.4 Considerações Finais

Com o intuito de validar o modelo de contexto e o filtro semântico proposto foi desenvolvido um estudo de caso. Neste estudo de caso foi proposto um cenário de aplicação onde um aluno está próximo ao prédio da biblioteca e recebe recomendações de livros que são de seu interesse. Para aplicar o cenário foi proposto um modelo genérico de regra SWRL capaz de filtrar os objetos de aprendizagem do ambiente.

Tal regra foi instanciada no modelo de contexto desenvolvido neste trabalho e o motor de inferência Pellet foi utilizado para raciocinar sobre a mesma. O motor de inferência Pellet foi escolhido pois foi o motor que apresentou o conjunto de características mais completo entre os motores estudados. Como resultado da inferência foi possível obter o par (objeto de aprendizagem recomendado, usuário) como era esperado.

Uma das vantagens ao utilizar o modelo ontológico e as regras SWRL é trazer toda a semântica da aplicação, que tradicionalmente fica no código fonte, para um nível conceitual mais alto, tornando o conhecimento do domínio e do comportamento da aplicação entendíveis por agentes computacionais e humanos.

No próximo capítulo são apresentadas as conclusões deste trabalho, os trabalhos futuros e os resultados alcançados através de publicações científicas do mesmo.

6 CONCLUSÃO

Campus universitários são ambientes complexos e dotados de recursos utilizados por pessoas. Contudo a quantidade de recursos disponível em um campus é tão grande que as pessoas podem acabar selecionando e utilizando um conjunto desses que não satisfazem suas necessidades e interesses da melhor maneira possível.

Um dos recursos mais disponíveis e utilizados no campus são os objetos de aprendizagem que acarretam outro problema consigo, já que além do usuário não ter a garantia que seus interesses serão correspondidos ele pode sentir-se cognitivamente cansado após ler e analisar uma quantidade considerável de objetos de aprendizagem.

Dentro deste contexto, torna-se interessante a utilização de um sistema de recomendação para filtrar os objetos de aprendizagem de interesse do usuário. Entretanto é interessante que o tipo de recomendação efetuada por esses sistemas leve em consideração informações a respeito do contexto do usuário e do objeto de aprendizagem a ser recomendado.

Uma das informações de contexto mais importantes e utilizadas em sistemas pervasivos é a informação de localização, tal informação tem sido utilizada principalmente na elaboração de recomendações para usuários de dispositivos móveis, já que esses podem ter sua localização facilmente recuperada através do dispositivo. Utilizar a localização do usuário para recomendar objetos de aprendizagem faz que o sistema apresente recomendações mais adequadas à atual situação já que ao invés de recomendar todos os objetos de aprendizagem de interesse, o sistema recomenda todos os objetos de aprendizagem de interesse que estão próximos ao usuário. Por conta disto, o número de objetos apresentado ao usuário é menor e mais relevante.

Outra característica a ser considerada um sistema de recomendação sensível ao contexto é encontrar uma forma de apresentar tais recomendações de maneira simples e adaptada ao usuário. Neste contexto, o desenvolvimento de uma interface adaptativa pode ajudar não somente a usabilidade mas a aceitação do sistema pelo usuário.

Para produzir tais recomendações este trabalho apresenta um modelo ontológico de contexto que utiliza quatro dimensões de informações e que tem o usuário como seu elemento central, as dimensões de informação são relativas ao perfil do usuário, a localização, a tecnologia e os objetos de aprendizagem a serem recomendados.

O modelo ontológico é instanciado com informações coletadas do ambiente, do bancos de dados da universidade e de repositórios de objetos de aprendizagem. Junto a esse modelo são construídas regras responsáveis pela filtragem dos objetos de aprendizagem, tais regras são escritas em forma de antecedente e consequente e utilizam os conceitos e relacionamentos do modelo para isso. A vantagem de construir as regras responsáveis pela recomendação dos objetos de aprendizagem é tornar a semântica de comportamento da aplicação explícita e inteligível tanto por humanos quanto por agentes computacionais.

Com o intuito de avaliar a eficácia da proposta, foi desenvolvido um estudo de caso com um cenário de aplicação e a atual execução das regras de recomendação. O motor de inferência Pellet foi utilizado para a execução das inferências lógicas sobre as regras

construídas. Os resultados obtidos através do estudo de caso mostraram que a abordagem é eficaz em realizar as recomendações propostas.

Como trabalhos futuros, pretende-se tornar a abordagem desenvolvida em protótipo e realizar testes com usuários no campus, com o intuito de avaliar a eficácia das recomendações. Outro trabalho futuro será o desenvolvimento de uma abordagem híbrida, que leve em consideração o modelo de contexto aqui desenvolvido e o algoritmo de filtragem colaborativa (mais tradicional algoritmo de sistemas de recomendação).

O objetivo de desenvolver tal abordagem híbrida é aproveitar-se das vantagens oferecidas por uma técnica para encobrir as desvantagens da outra. Como o problema da dificuldade na atualização dos dados, recebidos de feedbacks do usuário, no modelo ontológico e o problema da partida a frio dos algoritmos de filtragem colaborativa. Outro trabalho futuro é a utilização de técnicas para garantir a privacidade dos dados do usuário e a segurança entre os canais de comunicação.

REFERÊNCIAS

- ABBURU, S. A Survey on Ontology Reasoners and Comparison. **International Journal of Computer Applications**, New York, NY, USA, v. 57, n. 17, p. 33–39, nov. 2012.
- ADOMAVICIUS, G. et al. Incorporating contextual information in recommender systems using a multidimensional approach. **ACM Transactions on Information Systems**, New York, NY, USA v. 23, n. 1, p. 103–145, jan. 2005.
- ADOMAVICIUS, G.; TUZHILIN, A. Context-Aware Recommender Systems. In: RICCI, F. et al. (Eds.). **Recommender Systems Handbook SE - 7**. New York: Springer US, 2011. p. 217–253.
- AL-MEKHLAFI, K.; HU, X.; ZHENG, Z. **An Approach to Context-Aware Mobile Chinese Language Learning for Foreign Students** In: EIGHTH INTERNATIONAL CONFERENCE ON MOBILE BUSINESS., 8., 2009, Dalian. **Anais...** Dalian: IEEE, 2009. p. 340-346 Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5169283>>. Acesso em: 13 jan. 2014
- BAADER, F. **The description logic handbook: theory, implementation, and applications**. 1 ed. New York: Cambridge university press, 2003.
- BANAVAR, G. et al. **Challenges: An Application Model for Pervasive Computing** In: INTERNATIONAL CONFERENCE ON MOBILE COMPUTING AND NETWORKING - MOBICOM '00., 6., 2000, New York, **Anais...** New York: ACM Press, 2000, p. 266-274 Disponível em: <<http://doi.acm.org/10.1145/345910.345957>>. Acesso em: 30 set. 2013
- BECHHOFFER, S. et al. **OWL Web Ontology Language - Reference**. Standard, W3C World, [s.l.], OWL Working Group. p. 43, fev. 2004. Disponível em: <<http://www.w3.org/TR/owl-ref/>>. Acesso em: 08 out. 2013
- BETTINI, C. et al. A survey of context modelling and reasoning techniques. **Pervasive and Mobile Computing**, [s.l.] v. 6, n. 2, p. 161–180, abr. 2010.
- BOUZEGHOUB, A.; DO, K. N.; WIVES, L. K. **Situation-Aware Adaptive Recommendation to Assist Mobile Users in a Campus Environment** In: INTERNATIONAL CONFERENCE ON ADVANCED INFORMATION NETWORKING AND APPLICATIONS., 23., 2009, Washington, DC, USA **Anais...**: Washington, DC, USA: IEEE, 2009, 503-509. Disponível em: <<http://dx.doi.org/10.1109/AINA.2009.120>>. Acesso em: 7 set. 2013
- BRUSILOVSKY, P. Adaptive Hypermedia. **User Modeling and User-Adapted Interaction**, Dordrecht v. 11, n. 1-2, p. 87–110, mar. 2001.
- BRUSILOVSKY, P.; MAYBURY, M. T. From adaptive hypermedia to the adaptive web. **Communications of the ACM**, New York, v. 45, n. 5, p. 30–33, maio 2002.
- BRUSILOVSKY, P.; MILLÁN, E. User Models for Adaptive Hypermedia and Adaptive Educational Systems. In: BRUSILOVSKY, P.; KOBSA, A.; NEJDL, W. (Eds.). **The Adaptive Web Methods and Strategies of Web Personalization**. [s.l.]: Springer Berlin Heidelberg, 2007. p. 3–53.

- CHEN, H. et al. **SOUPA: standard ontology for ubiquitous and pervasive applications**, In: INTERNATIONAL CONFERENCE ON MOBILE AND UBIQUITOUS SYSTEMS: NETWORKING AND SERVICES, 1., 2004., Boston, **Anais...** Boston: IEEE, 2004, p. 258-267 Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1331732>>. Acesso em: 11 jan. 2014
- DENTLER, K. et al. Comparison of reasoners for large ontologies in the OWL 2 EL profile. **Semantic Web**, [s.l.], v. 2, n. 2, p. 71–87, jan. 2011.
- DEY, A. K. Understanding and Using Context. **Personal and Ubiquitous Computing**, London, v. 5, n. 1, p. 4–7, fev. 2001.
- DOURISH, P. What we talk about when we talk about context. **Personal and Ubiquitous Computing**, London, v. 8, n. 1, p. 19–30, fev. 2004.
- FERRIS, C. **Web services architecture**. Standard, W3C World, [s.l.], W3C Working Group, p. 98, fev. 2004. Disponível em: <<http://www.w3.org/TR/ws-arch/>>. Acesso em: 12 dez 2013
- GOLDBERG, D. et al. Using collaborative filtering to weave an information tapestry. **Communications of the ACM**, New York, v. 35, n. 12, p. 61–70, dez. 1992.
- GOLEMATI, M. et al. **Creating an ontology for the user profile: Method and applications**. In: INTERNATIONAL CONFERENCE ON RESEARCH CHALLENGES IN INFORMATION SCIENCE., 1., 2007. Ouarzazate **Anais...** Ouarzazate: IEEE, 2007, p. 407-412
- GRUBER, T. R. A Translation Approach to Portable Ontology Specifications. **Knowledge Acquisition**. London, v. 5, n. 2, p. 199–220, jun. 1993.
- GUO, Y.; PAN, Z.; HEFLIN, J. LUBM: A benchmark for OWL knowledge base systems. **Web Semantics: Science, Services and Agents on the World Wide Web**, [s.l.], v. 3, n. 2–3, p. 158–182, out. 2005.
- HERLOCKER, J. L. et al. Evaluating collaborative filtering recommender systems. **ACM Transactions on Information Systems**, New York, v. 22, n. 1, p. 5–53, jan. 2004.
- HERVÁS, R.; BRAVO, J. Towards the ubiquitous visualization: Adaptive user-interfaces based on the Semantic Web. **Interacting with Computers**, New York, v. 23, n. 1, p. 40–56, jan. 2011.
- HONG, M.; CHO, D. Ontology Context Model for Context-Aware Learning Service in Ubiquitous Learning Environments. **International Journal of Computers**, [s.l.], v. 2, n. 3, p. 193-200, jul. 2008.
- HORROCKS, I. et al. **SWRL: A Semantic Web Rule Language Combining OWL and RuleML**, Standard, W3C World, [s.l.], W3C Member Submission, p. 20, mai. 2004. Disponível em: <<http://www.w3.org/Submission/SWRL/>> Acesso em: 5 out 2013.
- INSTITUTE OF ELECTRICAL AND ELECTRONIC ENGINEERING **Standard for Learning Object Metadata**. **IEEE Std 1484.12.1-2002**, New York, 2002.
- INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Pesquisa Nacional por Amostra de Domicílios - Acesso à Internet e Posse de Telefone Móvel Celular Para Uso Pessoal**. Rio de Janeiro: [s.n.]. Disponível em:

<ftp://ftp.ibge.gov.br/Acesso_a_internet_e_posse_celular/2011/PNAD_Inter_2011.pdf>
 . Acesso em: 7 jan. 2014

INSTITUTO BRASILEIRO DE OPINIÃO PÚBLICA E ESTATÍSTICA. **Um Novo Cenário para o Consumo de Mídia**. [s.l.: s.n.], nov. 2013. Disponível em:
 <<http://www.ibope.com.br/pt-br/conhecimento/Infograficos/Paginas/Um-novo-cenario-para-o-consumo-de-midia.aspx>>. Acesso em: 28 jan. 2014.

INTERNATIONAL TELECOMMUNICATION UNION. **Yearbook of Statistics 2012**. [s.l.: s.n.], 38. ed., 2012.

LI, M. et al. **Personalization in Context-aware Ubiquitous Learning-Log System**. In: International Conference on Wireless, Mobile and Ubiquitous Technology in Education., 7., 2012, Takamatsu **Anais...** Takamatsu: IEEE, mar. 2012, p. 41-48.
 Disponível em:
 <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6185078>>. Acesso em: 18 set. 2013

NIELSEN, J. **Usability Inspection Methods**. In: CONFERENCE COMPANION ON HUMAN FACTORS IN COMPUTING SYSTEMS, 13., Boston. **Anais...** Boston: ACM, Inc., 1994. p. 413-414.

NOY, N. F.; MCGUINNESS, D. L. **Ontology development 101: A guide to creating your first ontology**. [s.l.], Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880, p. 25, 2001.

PANTSAR-SYVANIEMI, S.; SIMULA, K.; OVASKA, E. **Context-awareness in smart spaces**. In: SYMPOSIUM ON COMPUTERS AND COMMUNICATIONS, 15., 2010, Riccione. **Anais...** Riccione: IEEE, jun. 2010. p. 1023-1028. Disponível em:
 <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5546630>. Acesso em: 14 jan. 2014

PERNAS, A. M. et al. Enriching adaptation in e-learning systems through a situation-aware ontology network. **Interactive Technology and Smart Education**, [s.l.], v. 9, n. 2, p. 60–73, ago. 2012.

RICCI, F.; ROKACH, L.; SHAPIRA, B. Introduction to Recommender Systems Handbook. In: RICCI, F. et al. (Eds.). **Recommender Systems Handbook SE - 1**. [s.l.] Springer US, 2011. p. 1–35.

SARWAR, B. et al. **Item-based collaborative filtering recommendation algorithms**. In: INTERNATIONAL CONFERENCE ON WORLD WIDE WEB - WWW '01, 10., 2001, New York. **Anais...** New York: ACM Press, 2001. p. 285-295. Disponível em:
 <<http://doi.acm.org/10.1145/371920.372071>>. Acesso em: 8 nov. 2013

SCHILIT, B.; ADAMS, N.; WANT, R. **Context-Aware Computing Applications**. In: WORKSHOP ON MOBILE COMPUTING SYSTEMS AND APPLICATIONS, 1., 1994, Washington. **Anais...** Washington: IEEE, 1994. p. 85-90. Disponível em:
 <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4624429>>. Acesso em: 21 set. 2013

SETTEN, M.; POKRAEV, S.; KOOLWAAIJ, J. Context-Aware Recommendations in the Mobile Tourist Application COMPASS. In: BRA, P. E.; NEJDJL, W. (Eds.). **Adaptive Hypermedia and Adaptive Web-Based Systems SE - 27**. [s.l.]: Springer Berlin Heidelberg, 2004. v. 3137 p. 235–244.

THOMAS, G. A typology for the case study in social science following a review of definition, discourse, and structure. **Qualitative Inquiry**, [s.l.], v. 17, n. 6, p. 511–521, jun. 2011.

WANG, S.-L.; WU, C.-Y. Application of context-aware and personalized recommendation to implement an adaptive ubiquitous learning system. **Expert Systems with Applications**, Tarrytown, v. 38, n. 9, p. 10831–10838, set. 2011.

WANG, X. H. et al. **Ontology based context modeling and reasoning using OWL**. In: CONFERENCE ON PERVASIVE COMPUTING AND COMMUNICATIONS WORKSHOPS, 2., 2004, Orlando. **Anais...** Orlando: IEEE, 2004. p. 18-22. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1276898>>. Acesso em: 11 jan. 2014

WEISER, M. The computer for the 21st century. **ACM SIGMOBILE Mobile Computing and Communications Review**, New York, v. 265, n. 3, p. 94–104, jul. 1991.

YANG, S. J. H. Context aware ubiquitous learning environments for peer-to-peer collaborative learning. **Educational Technology & Society**, [s.l.], v. 9, n. 1, p. 188–201, jan. 2006.