

**Um Ambiente Virtual Inteligente e
Adaptativo Baseado em
Modelos de Usuário e Conteúdo**

por

CÁSSIA TROJAHN DOS SANTOS

Dissertação de mestrado submetida à avaliação
como requisito parcial para obtenção do grau de
Mestre em Computação Aplicada

Prof. Dr. Fernando Santos Osório
Orientador

São Leopoldo, fevereiro de 2004.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Santos, Cássia Trojahn dos

Um Ambiente Virtual Inteligente e Adaptativo Baseado em Modelos de Usuário e Conteúdo / por Cássia Trojahn dos Santos. – São Leopoldo: Ciências Exatas e Tecnológica, UNISINOS, 2004.

131 f.: il.

Dissertação (mestrado) – Universidade do Vale do Rio dos Sinos. Ciências Exatas e Tecnológicas. Programa Interdisciplinar de Pós-Graduação em Computação Aplicada – PIPCA, São Leopoldo, BR – RS, 2004. Orientador: Osório, Fernando Santos.

I. Osório, Fernando Santos. II. Título.

UNIVERSIDADE DO VALE DO RIO DOS SINOS

Reitor: Dr. Aloysio Bohnen

Vice-Reitor: Marcelo Fernandes de Aquino

Pró-Reitor Acadêmico: Padre Pedro Gilberto Gomes

Unidade Acadêmica de Pós-Graduação e Pesquisa: Ione Bentz

Coordenador do PIPCA: Prof. Dr. Arthur Tórgo Gómez

Aos meus pais amados.

Agradecimentos

Aos meus pais, Celso e Iria, e irmãos, Cassandra e Mano, pessoas que amo e admiro, especialmente à minha mãe, fonte inesgotável de entusiasmo, energia e conforto. Obrigada por me oportunizarem o estudo, a maior herança que poderiam ter me oferecido.

Aos meus avós, tios, tias, primos e primas, pelos domingos alegres, pelo apoio imensurável e por me fazerem constatar que a família é o bem mais precioso que se pode ter.

Às grandes amigas conquistadas, Cláudia Pérez, Sandra Collovini e Cassiana Silva, com as quais convivi, quase que diariamente, ao longo destes dois anos. Compartilhamos não apenas o ambiente de trabalho, mas também as dificuldades, as incertezas e as conquistas. Um agradecimento especial ao pessoal do Laboratório da Engenharia da Linguagem, Rodrigo Goulart, Fábio Okuyama, Douglas Silva, Regina Martins e César Coelho, pelas ajudas e por tornar o nosso ambiente de trabalho extremamente agradável e alegre.

Ao Eduardo Battistella e à Kátia Hardt, pelo companheirismo e amizade.

Ao Prof. Fernando Osório, pela orientação, conhecimento compartilhado e paciência durante todo o desenvolvimento da dissertação. Um agradecimento especial, à Profa. Renata Vieira, por me conduzir no estágio docência e acolher junto ao seu grupo. Exemplos de profissionalismo que tomarei como referências.

Às Profas. Lúcia Giraffa e Soraia Musse, pela disposição em acompanhar o trabalho e pelas valiosas contribuições.

Aos professores e funcionários do Programa Interdisciplinar de Pós-Graduação em Computação Aplicada, pelo conhecimento compartilhado e pelas ajudas, e à Capes, por viabilizar o meu mestrado.

Finalmente, agradeço imensamente a Deus, por me acompanhar sempre e por ter colocado pessoas formidáveis em minha vida.

Resumo

A Realidade Virtual (RV) tornou-se uma alternativa atraente para o desenvolvimento de interfaces visuais mais realistas e interessantes para o usuário. Os ambientes que fazem uso de técnicas de RV são comumente referenciados como Ambientes Virtuais (AVs). Atualmente, atenção tem sido atribuída à integração de Inteligência Artificial (IA) e AVs. O objetivo é obter maior usabilidade e realismo das interfaces, explorando a combinação de objetos tridimensionais e entidades inteligentes. Os ambientes que exploram tal integração são chamados Ambientes Virtuais Inteligentes (AVIs). Esta dissertação apresenta o AdapTIVE (*Adaptive Three-dimensional Intelligent and Virtual Environment*), um ambiente virtual que tem a sua estrutura e apresentação adaptadas de acordo com os interesses e as preferências dos usuários (representadas em um modelo de usuário) e conforme a manipulação (inserção, remoção ou atualização) de conteúdos no ambiente. Um processo de categorização automático de conteúdos é aplicado na criação de modelos de conteúdos, utilizados na organização espacial dos mesmos no ambiente. No processo de adaptação, os modelos de usuário e de conteúdo são utilizados. Além disso, um agente virtual inteligente atua como assistente do usuário na navegação pelo ambiente e na localização de informações relevantes. Para validar o ambiente proposto, um protótipo, envolvendo um ambiente de apoio ao ensino a distância, para a disponibilização de conteúdos, foi desenvolvido. A motivação para o desenvolvimento deste ambiente fundamentou-se nas seguintes premissas: dinamicidade dos ambientes de apoio a EaD (atualizações contínuas de conteúdos), diversidade de modelos de usuários; e uso promissor dos AVIs, permitindo a criação de ambientes altamente interativos.

Abstract

Virtual Reality (VR) becomes an attractive alternative for the development of more realistic and interesting visual interfaces for the user. The environments that make use of VR techniques are referred as Virtual Environments (VEs). Nowadays, the attention of research community has been focused to the integration of Artificial Intelligence (AI) and VEs. The objective is to create VEs that explore the use of intelligent entities and effective means for their graphical representations, together with interaction of different forms, improving dynamicity, realism and usability of these interfaces. The environments that explore such integration are called Intelligent Virtual Environments (IVEs). This dissertation presents the AdapTIVE (*Adaptive Three-dimensional Intelligent and Virtual Environment*), a virtual environment which has its structure and presentation customized according to users' interests and preferences (represented in a user model) and in accordance with insertion and removal of contents in this environment. An automatic content categorization process is applied in the creation of content models, used in the spatial organization of the content in the environment. An intelligent agent, who assists users in order to help to explore the environment and to locate information, is presented. In order to validate our proposal, a prototype of a distance learning environment, used to make educational content available, was developed. The motivation to the development of this environment for distance learning is based on the following main ideas: the dynamic essence of this type of environment (continuously updating contents); diversity of user models; and the promising use of IVEs, allowing the creation of highly interactive environments.

Sumário

Resumo	v
Abstract	vi
Lista de Tabelas	ix
Lista de Figuras	x
Lista de Abreviaturas	xii
1 Introdução	1
<i>1.1 Objetivos</i>	3
1.1.1 Objetivo Geral	3
1.1.2 Objetivos Específicos	3
<i>1.3 Organização do Texto</i>	4
2 Modelagem de Usuários	6
<i>2.1 Modelo de Usuário</i>	6
<i>2.2 Processo de Modelagem</i>	8
<i>2.3 Técnicas de IA na Modelagem de Usuários</i>	11
<i>2.4 Trabalhos Relacionados</i>	16
3 Modelagem de Conteúdos	21
<i>3.1 Modelo de Conteúdo</i>	23
<i>3.2 Processo de Modelagem</i>	24
<i>3.3 Trabalhos Relacionados</i>	26
<i>3.4 Metodologia</i>	28
<i>3.5 Experimentos Preliminares</i>	32
4 Agentes Inteligentes	37
<i>4.1 Definições</i>	37
<i>4.2 Características</i>	38
<i>4.3 Classificação</i>	40
<i>4.4 Arquiteturas</i>	43
4.4.1 Arquitetura Reativa	43
4.4.2 Arquitetura Cognitiva	44
4.4.3 Arquitetura Híbrida	45
4.4.4 Arquitetura Baseada em Estados Mentais	45

4.4.5 Outras Arquiteturas	46
4.5 Aplicações.....	47
5 Ambientes Virtuais Inteligentes	49
5.1 Trabalhos Relacionados.....	51
6 Ambiente Virtual Inteligente e Adaptativo.....	60
6.1 Contexto do Trabalho no Estado da Arte.....	60
6.2 Arquitetura AdapTIVE.....	62
6.2.1 Gerenciador de Modelos de Usuários.....	64
6.2.2 Gerenciador de Conteúdos	68
6.2.3 Gerador de Ambientes	69
6.2.4 Agente Virtual Inteligente	70
7 Protótipo.....	74
7.1 Estrutura do Ambiente.....	75
7.2 Gerenciador de Modelos de Usuário	77
7.3 Gerenciador de Conteúdos.....	82
7.4 Agente Virtual Inteligente.....	87
7.5 Extensão da Estrutura Inicial.....	97
8 Conclusão	100
Bibliografia Referenciada	106
Bibliografia Consultada	118

Lista de Tabelas

Tabela 2.1 – Tipos de informações e propriedades de um modelo de usuário.....	8
Tabela 3.1 – Erro de classificação (ADs).....	32
Tabela 3.2 – Erro de classificação (RPs).....	32
Tabela 3.3 – Média do menor erro de generalização (BP).....	34
Tabela 3.4 – Média do menor erro de generalização (CasCor).....	34
Tabela 3.5 – Média do menor erro de generalização.....	35
Tabela 4.1 – Classificações de agentes inteligentes.	42
Tabela 5.1 – Comparativo entre os AVIs.	59
Tabela 7.1 – Comparativo entre os categorizadores binário e múltiplo.	85
Tabela 7.2 – Opções de solicitações e respostas correspondentes.	87

Lista de Figuras

Figura 3.1 – Processo de categorização.....	24
Figura 3.2 – Etapas de pré-processamento e geração dos scripts, conforme a metodologia proposta.	31
Figura 4.1 – Representação genérica de um agente [Russel and Norvig, 1995].	38
Figura 5.1 – Interfaces de ambientes do Active Worlds.....	51
Figura 5.2 – (a) STEVE ao lado de um equipamento; (b) STEVE demonstrando como utilizar o equipamento [Rickel and Johnson, 1997].	52
Figura 5.3 – Agente Ulysses caminhando pelo ambiente [Bersot et al, 1998].....	52
Figura 5.4 – (a) Interface de entrada ao ambiente; (b) Agente se deslocando no ambiente [Panayiotopoulos et al, 1999].	53
Figura 5.5 – (a) Entrada do teatro virtual; (b) Agente virtual [Nijholt and Hulstijn 2000].	54
Figura 5.6 – Apresentação do produto pelo agente [Milde, 2000].	54
Figura 5.7 – Interação entre os agentes cliente e bibliotecário [Anastassakis et al, 2001].	55
Figura 5.8 – Sala de aula virtual [Rizzo et al, 2002].	56
Figura 5.9 – Várias configurações do escritório virtual [Rizzo et al, 2002].....	56
Figura 5.10 – Adaptações na loja virtual [Chittaro and Ranon, 2002; 2002a].	57
Figura 5.11 – (a) Agente explanando sobre o objeto; (b) Trajetória elaborada para a navegação pelo ambiente [Chittaro et al, 2003].	57
Figura 6.1 – Arquitetura AdapTIVE.	63
Figura 6.2 – Arquitetura do agente virtual inteligente.....	71
Figura 7.1 – Salas do ambiente.....	76
Figura 7.2 – Sub-salas do ambiente.....	76
Figura 7.3 – (a) Ambiente 3D com salas, vazias e com sub-salas; (b) Mapa 2D correspondente.....	77
Figura 7.4 – (a) Ambientes com salas com sub-salas; (b) Mapa 2D correspondente....	77
Figura 7.5 – Formulário para a coleta dos dados para composição do modelo inicial... 78	
Figura 7.6 – Usuário com interesse em IA.	79
Figura 7.7 – Usuário com interesse em CG.....	79
Figura 7.8 – Número de solicitações (S), navegações (N) e acesso (A).....	80
Figura 7.9 – FCs correspondentes a área de ES, ao longo das 7 sessões.	81
Figura 7.10 – (a) Organização do ambiente de acordo com o modelo inicial do usuário; (b) organização do ambiente após alterações no modelo do usuário.	81

Figura 7.11 – (a) Organização do ambiente de acordo com o modelo inicial do usuário; (b) Organização do ambiente após alterações no modelo do usuário.....	82
Figura 7.12 – Representação de conteúdos no ambiente.....	83
Figura 7.13 – Visualização de um conteúdo, após clique na descrição correspondente.	83
Figura 7.14 – Interface para definição do modelo de um conteúdo.	84
Figura 7.15 – (a) Árvore gerada a partir de um categorizador múltiplo (área RC); (b) Árvore gerada com um categorizador binário (área RC e sub-área Protocolos)....	86
Figura 7.16 – Solicitação do usuário para a localização da área de IA.....	88
Figura 7.17 – Localização, pelo agente, da área solicitada.	88
Figura 7.18 – Solicitação do usuário para a localização da sub-área de RNA.....	89
Figura 7.19 – Localização, pelo agente, da sub-área solicitada.	89
Figura 7.20 – Indicação das palavras-chave para a recuperação de conteúdos.	90
Figura 7.21– Listagem dos títulos e sub-salas correspondentes aos conteúdos recuperados.....	90
Figura 7.22 – Agente apresentando o ambiente ao usuário.....	91
Figura 7.23 – Interação para a inserção de um conteúdo no ambiente.....	92
Figura 7.24 – Formulário para a especificação do modelo do conteúdo.....	93
Figura 7.25 – Agente conduzindo usuário até a sub-sala onde o conteúdo foi inserido.	93
Figura 7.26 – Agente apresentando a sub-sala onde o conteúdo foi inserido.	94
Figura 7.27 – Provedor visualizando o objeto que representa o conteúdo adicionado. .	94
Figura 7.28 – Mapa topológico do ambiente.....	95
Figura 7.29 – Ambiente 3D e mapa 2D correspondente.	96
Figura 7.30 – (a) Solicitação do usuário ao agente; (b) localização de uma área pelo agente; (c) visualização de conteúdos da sub-área RNAs (IA); (d) visualização de detalhes de um conteúdo, após clique em uma descrição de conteúdo.	97
Figura 7.31 – Ambiente com as novas áreas adicionadas.	98
Figura 7.32 – Mapa 2D do ambiente.	98

Lista de Abreviaturas

2D	Bidimensional
3D	Tridimensional
AD	Árvore de Decisão
AEIO	<i>Agents, Environment, Interaction and Organization</i>
A-NN	<i>Algorithm Nearest Neighbor</i>
AG	Algoritmo Genético
AV	Ambiente Virtual
AVI	Ambiente Virtual Inteligente
BDI	<i>Belief Desire Intention</i>
BP	<i>BackPropagation</i>
CasCor	<i>Cascade Correlation</i>
CG	Computação Gráfica
EaD	Educação a Distância
ES	Engenharia de Software
FC	Fator de Certeza
HMD	<i>Head Mounted Display</i>
HTML	<i>HyperText Markup Language</i>
IA	Inteligência Artificial
IDF	<i>Inverse Document Frequency</i>
IHC	Interface Homem Computador
KSI	<i>Knowledge, Status and Intention</i>
LOM	<i>Learning Object Metadata</i>
MC	Medida de Crença
MD	Medida de Descrença
MLP	<i>Multi-Layer Perceptron</i>
PLN	Processamento de Linguagem Natural
RC	Rede de Computador
RI	Recuperação de Informação
RNA	Rede Neural Artificial
RP	Regra de Produção

RV	Realidade Virtual
SI	Sistema de Informação
SMA	Sistema Multiagente
SOM	<i>Self-Organizing Maps</i>
STI	Sistema Tutor Inteligente
TF	<i>Term Frequency</i>
VA	Vida Artificial
VRML	<i>Virtual Reality Modeling Language</i>
XML	<i>eXtensible Markup Language</i>

1 Introdução

A Realidade Virtual (RV) tornou-se uma alternativa atraente para o desenvolvimento de interfaces visuais mais realistas e interessantes para o usuário [Teichrieb, 1999]. Os ambientes que fazem uso de técnicas de RV são comumente referenciados como Ambientes Virtuais (AVs).

Em AVs, conforme Avradinis et al (2000), o usuário é uma parte do sistema, uma presença autônoma no ambiente. Ele está livre para navegar, interagir com objetos e examinar o ambiente de diferentes pontos de vista, o que não é possível em ambientes bidimensionais. Segundo Frery et al (2002), o paradigma 3D é útil por oferecer a possibilidade de representar a informação de um modo realístico, organizando-a de uma maneira espacial. Deste modo, obtém-se maior intuição na visualização da informação, permitindo ao usuário explorá-la de um modo interativo, mais natural ao ser humano. Além disso, a possibilidade de interação entre múltiplos usuários nesses ambientes, pode satisfazer as necessidades sociais dos usuários.

Atualmente, atenção tem sido atribuída à integração de Inteligência Artificial (IA) e AVs. O objetivo é obter maior usabilidade e realismo das interfaces, explorando a combinação de objetos tridimensionais e entidades inteligentes. De acordo com Aylett and Luck (2000) e Aylett and Cavazza (2001), os ambientes que exploram tal integração são chamados Ambientes Virtuais Inteligentes (AVIs) e têm derivado uma nova área de pesquisa.

Conforme Anastassakis et al (2001), um AVI é um ambiente virtual semelhante a um mundo real, habitado por entidades autônomas inteligentes exibindo uma variedade de comportamentos. Estas entidades podem ser objetos estáticos simples ou dinâmicos, representações virtuais de formas de vida (humanos ou animais), avatares (que representam usuários no ambiente), entre outros. Segundo Rickel et al (2002) e Gratch et al (2002), as aplicações potenciais desses ambientes são consideráveis, podendo ser empregados em uma variedade de áreas, principalmente relacionadas com a simulação, entretenimento, educação e treinamento. Em simulação, ambientes de diferentes tipos (espaços urbanos abertos ou interiores, habitados por humanos virtuais)

podem auxiliar em projetos arquitetônicos e no controle de tráfego de pessoas ou carros. Na área de entretenimento, os jogos (com cenários que podem ser adaptados conforme o andamento do jogo), os teatros e as lojas virtuais (onde o usuário pode navegar e interagir com outros usuários) podem ser citados como aplicações potenciais. Por fim, em sistemas educacionais, a incorporação de personagens tutores e sofisticadas técnicas de representações da informação provêm uma efetiva e agradável experiência de aprendizado.

Neste contexto, três aspectos podem ser destacados. Primeiro, uma interação mais efetiva pode ser obtida a partir da adaptação do ambiente, conforme interesses e preferências de usuários individuais ou de grupos de usuários. Conforme Chittaro and Ranon (2002a), a capacidade de (semi)automaticamente adaptar o conteúdo, estrutura e/ou apresentação do ambiente, de acordo com as características dos usuários é cada vez mais considerado um fator chave para incrementar o nível de satisfação dos usuários. Em um shopping virtual, por exemplo, a possibilidade de alterar a disposição das lojas, conforme os interesses do usuário (impossível de ser feita em um shopping real, e pouco efetiva em um ambiente bidimensional) pode tornar a interface mais usável e atrativa.

Segundo, a organização espacial das informações em um ambiente tridimensional exige, em muitos casos, o agrupamento das mesmas, de acordo com algum critério semântico. Por exemplo, em uma loja virtual é interessante o agrupamento dos produtos conforme a seção a que pertencem (eletrônicos, bazar, outros) ou, em um ambiente de apoio a Educação a Distância (EaD), é necessário agrupar espacialmente os conteúdos conforme a área a que pertencem. Desta forma, o uso de ferramentas que auxiliem na organização espacial das informações no ambiente se faz necessário.

Terceiro, de acordo com Ballegooij and Eliëns (2001), os ambientes virtuais tridimensionais sofrem com o problema dos usuários sentirem-se “perdidos” no ambiente. Deste modo, recursos para prover assistência aos usuários na navegação pelo ambiente e localização de informações são necessários. Dentre os recursos geralmente utilizados, podem ser citados: a identificação de locais no ambiente (uso de setas e placas); os mapas 2D do ambiente; e os assistentes virtuais. Conforme Chittaro and Ranon (2003), a introdução de assistentes virtuais tem a vantagem adicional de tornar o ambiente mais natural e atrativo para o usuário. Assim, o uso de assistentes virtuais tem

as seguintes vantagens: enriquece a interação com o ambiente virtual [Rickel et al, 2002]; torna o ambiente menos intimidador e mais natural ao usuário [Chittaro et al, 2003]; evita que os usuários sintam-se perdidos no ambiente [Rickel and Johnson, 2000].

Esta dissertação de mestrado endereça os três aspectos comentados acima, propondo o ambiente AdapTIVE (*Adaptive Three-dimensional Intelligent and Virtual Environment*), um ambiente virtual que tem a sua estrutura e apresentação adaptadas de acordo com os interesses e as preferências dos usuários (representadas em um modelo de usuário) e conforme a manipulação (inserção, remoção ou atualização) de conteúdos no ambiente. Um processo de categorização automático de conteúdos é aplicado na criação de modelos de conteúdos, utilizados na organização espacial dos mesmos no ambiente. No processo de adaptação, os modelos de usuário e conteúdo são utilizados. Além disso, um agente virtual inteligente atua como assistente dos usuários na navegação pelo ambiente e na localização de informações relevantes. Para validar o ambiente proposto, um protótipo, envolvendo um ambiente de apoio ao ensino a distância para a disponibilização de conteúdos, foi desenvolvido.

1.1 Objetivos

1.1.1 Objetivo Geral

Desenvolver um ambiente virtual inteligente e adaptativo que utiliza modelos de usuários e de conteúdos no processo de adaptação, possuindo um agente inteligente que assiste os usuários na navegação pelo ambiente, fornecendo informações para a localização de conteúdos relevantes.

1.1.2 Objetivos Específicos

- Definir um modelo inicial de usuário que possa ser alterado em decorrência das interações com o ambiente e com o agente;
- Definir e implementar um mecanismo para a atualização do modelo inicial de usuário;

- Modelar e implementar uma ferramenta para aquisição e representação de um modelo de conteúdo, baseada no processo de categorização automático de conteúdos textuais;
- Modelar e implementar uma ferramenta para auxílio à manipulação (inserção, remoção e atualização) de conteúdos no ambiente;
- Modelar e implementar um agente virtual inteligente (assistente do usuário);
- Definir e implementar um protocolo de comunicação entre os usuários e o agente, que suporte o processo de interação de maneira ágil e eficiente;
- Modelar e implementar uma ferramenta para a construção de ambientes adaptados, conforme modelos de usuário e conteúdo;
- Propor uma arquitetura que integre os componentes acima listados.

1.3 Organização do Texto

Esta dissertação possui aporte teórico focado em quatro áreas de pesquisa: modelagem de usuários, modelagem de conteúdos, agentes inteligentes e ambientes virtuais inteligentes. O texto está organizado como segue.

No capítulo 2, é apresentada uma visão geral da área de modelagem de usuários. São comentadas as definições para um modelo de usuário, é descrito o processo de modelagem e são apresentadas as abordagens de IA aplicadas ao processo. Ao final, os trabalhos relacionados à modelagem de usuário são apresentados.

A modelagem de conteúdos é abordada no capítulo 3. São apresentadas as definições para um modelo de conteúdos e é comentado o processo de modelagem de conteúdos, como uma tarefa associada a categorização de conteúdos. Os trabalhos que fazem uso de um processo de categorização de textos e a metodologia para o processo proposto nesta dissertação são apresentados. Ao final, os experimentos preliminares realizados com a aplicação de técnicas de aprendizado de máquina são comentados.

No capítulo 4, é apresentada uma introdução à abordagem de agentes inteligentes. São apresentadas as principais propriedades que caracterizam um agente inteligente, bem como as classificações, arquiteturas e aplicações destes agentes.

No capítulo 5, é dada uma visão geral sobre os AVIs. É feita uma introdução aos ambientes e são expostos os trabalhos relacionados à combinação desses e agentes inteligentes.

O capítulo 6 aborda o ambiente AdapTIVE. É feita a contextualização do trabalho no estado da arte, onde são enfatizadas as contribuições do mesmo em relação aos assuntos e trabalhos revisados nos capítulos anteriores. A arquitetura do ambiente é apresentada e cada um de seus componentes é detalhado.

No capítulo 7 é apresentado o protótipo desenvolvido para validar a proposta. No capítulo 8, são apresentadas as considerações finais e as propostas para trabalhos futuros.

2 Modelagem de Usuários

Uma das principais motivações em se construir sistemas capazes de modelar o usuário está na possibilidade de adaptar o comportamento do sistema conforme as necessidades particulares de seus usuários [Papatheodorou, 2001]. Um modelo de usuário é uma representação explícita das características, preferências e necessidades de um usuário ou grupo de usuários. Ao processo de aquisição e representação de um modelo dá-se o nome de modelagem de usuário.

Historicamente, a modelagem de estudantes antecede a modelagem de usuários [Webb et al, 2001]. A área de Sistemas Tutores Inteligentes (STIs), no final da década de 70, foi a primeira a considerar as características individuais dos usuários, como forma de prover uma instrução individualizada. O termo modelagem de usuário passou a ser considerado apenas no final da década de 80 [Self, 1999].

Desde então, diversas abordagens e técnicas para a modelagem de usuários vêm sendo propostas em pesquisas nas áreas de Interação Homem-Computador (IHC), Recuperação de Informação (RI) e Inteligência Artificial (IA). E como consequência disto, aplicações que colecionam diferentes tipos de informações sobre os seus usuários e exibem diferentes formas de adaptação têm sido desenvolvidas.

Este capítulo apresenta conceitos básicos relativos à modelagem de usuários e está organizado como segue. Na seção 2.1, é feita uma revisão sobre os modelos de usuários. A seção 2.2 apresenta o processo de modelagem, enfatizando as etapas de aquisição e representação de um modelo. Na seção 2.3, são comentadas as abordagens de IA aplicadas ao processo de modelagem. Por fim, a seção 2.4 apresenta os trabalhos relacionados a modelagem de usuários.

2.1 Modelo de Usuário

Um modelo de usuário é uma coleção de informações e suposições sobre usuários individuais ou sobre grupos de usuários, necessárias para que o sistema adapte diversos aspectos de suas funcionalidades e interface [Kobsa, 1995]. Conforme Abbattista et al

(2002), compreende toda a informação sobre o usuário, extraída da sua interação com o sistema.

Ele representa o conhecimento sobre o usuário do sistema, utilizado com o propósito de melhorar a interação [Fischer, 2001]. Consiste, principalmente, de conhecimento sobre as preferências individuais do usuário, as quais determinam o comportamento do mesmo [Papatheodorou, 2001]. O termo perfil de usuário [Wasfi, 1999] também é utilizado com o propósito de representar o conhecimento sobre o usuário.

Existe um grande número de informações sobre o usuário que pode ser incluído em um modelo. Kass and Finin (1988) consideram quatro tipos: objetivos, capacidades, comportamentos, e conhecimento ou crenças. Horvitz et al (1998) incluem, ainda, as necessidades, bem como os objetivos, definindo objetivos como as tarefas ou sub-tarefas que recebem a atenção do usuário, e necessidades como a ação automatizada ou a informação que irá reduzir o tempo ou esforço requerido para atingir o objetivo. Além disso, podem ser incluídos os interesses e as preferências do usuário.

Quanto as propriedades de um modelo, Kass and Finin (1988) indicam as seguintes:

- *especialização*: um modelo pode ser genérico (refletindo as propriedades de um grupo) ou individual;
- *modificabilidade*: um modelo pode ser dinâmico (ser alterado durante o curso da interação) ou estático;
- *extensão temporal*: um modelo pode abranger os interesses de curto prazo (*short-term*) do usuário, de longo prazo (*long-term*), ou ambos. Um modelo que reflete os interesses de curto prazo é formado, basicamente, pelas informações mais recentemente coletadas, enquanto os de longo-prazo refletem as propriedades gerais do usuário. Os modelos estáticos são essencialmente os que refletem os interesses de longo prazo;
- *método de uso*: um modelo pode ser descritivo (descritos em uma base de dados) ou prescritivo (sistema interpreta os comportamentos do usuário, extraindo informações para compor o modelo).

A Tabela 2.1 apresenta uma síntese dos tipos de informações que podem estar contidos em um modelo de usuário e das propriedades utilizadas para caracterizá-lo.

Tabela 2.1 – Tipos de informações e propriedades de um modelo de usuário.

Propriedade	Valor
Tipo de informação	objetivos, capacidades, comportamentos, crenças, necessidades, interesses e preferências
Especialização	individual ou grupo
Modificabilidade	estático ou dinâmico
Extensão temporal	curto prazo ou longo prazo
Método de uso	descritivo ou prescritivo

2.2 Processo de Modelagem

O processo de modelagem de usuários envolve a coleta das informações necessárias para a elaboração do modelo e a representação das informações coletadas. Esse processo constitui a metodologia de modelagem e envolve definir o que deve ser representado e como a informação será representada [Abbattista et al, 2002]. Deste modo, pode ser descrito em termos das seguintes etapas: (a) identificação dos propósitos da utilização do modelo; (b) definição das propriedades que deverão compor o modelo; (c) escolha do formato de representação do modelo; (d) definição dos dados que deverão ser coletados, métodos e técnicas utilizadas para a coleta; (e) coleta dos dados; e (f) representação dos dados em um modelo de usuário.

A identificação dos propósitos da utilização do modelo deve considerar quais as tarefas desempenhadas pelo sistema, os objetivos do usuário no uso do mesmo e os aspectos do sistema que serão adaptados. Em seguida, são definidas as propriedades do modelo: especialização (individual ou de grupo), extensão temporal (curto ou longo prazo), tipo de informação (interesses, preferências, por exemplo), outras. A etapa seguinte consiste em definir o formato de representação do modelo, podendo ser utilizados os seguintes formatos: conjuntos de preferências, predicados em lógica de primeira ordem ou grupos de predicados, planos, árvores de decisões, redes neurais, tabelas em bases de dados ou como vetor de palavras em um hiper-espaço, outros.

A seguir, devem ser determinados os tipos de dados que deverão ser coletados, de forma que as propriedades relevantes do usuário possam ser extraídas, bem como são

selecionados os métodos e técnicas utilizadas para a coleta dos dados. Quanto aos tipos de dados que podem ser coletados, eles são classificados [Adomavicius and Tuzhilin, 2001; Abbattista et al, 2002] em: factuais (o que o usuário é, tais como: nome, gênero, idade, preferências), transacionais (compras efetuadas em um dado período, valores gastos, outros), navegacionais (páginas visitadas, tempo de permanência em cada página, outros) e demográficos (endereço, salário, ocupação, outros).

A próxima etapa consiste na coleta dos dados, propriamente dita, e na representação explícita dos dados coletados em um modelo de usuário. Johansson (2002) comenta que a utilidade de um modelo de usuário é dependente da relevância e da exatidão da informação que o modelo possui, as quais estão intimamente relacionadas à coleta dos dados, a partir dos quais o modelo será produzido.

Diversos métodos para coleta de dados têm sido apresentados na literatura. Em geral, eles podem ser agrupados em dois conjuntos [Pazzani and Billsus, 1997; Papatheodorou, 2001]: explícitos e implícitos.

Os métodos explícitos coletam as informações diretamente do usuário, por exemplo, questionando-o sobre os seus interesses, preferências e necessidades. Este tipo de coleta, geralmente, é feito através do uso de formulários que requisitam informações típicas, tais como gênero e ano de nascimento, e algumas informações específicas, tais como interesses e preferências (em um sistema de recomendação de produtos, por exemplo, poderão ser solicitadas as categorias de interesse). Considerando que uma limitada quantidade de informação pode ser adquirida desta forma (usuários podem ficar incomodados por preencherem formulários extensos ou por proverem informações pessoais e interesses), a abordagem usualmente aplicada é apresentar ao usuário apenas um conjunto limitado de campos e deixar ele decidir quais serão preenchidos.

Segundo Schwab and Kobsa (2002), os usuários não são, muitas vezes, motivados a informar explicitamente seus dados pessoais e, deste modo, as conclusões sobre os interesses do usuário não deveriam estar baseadas somente nas avaliações dos mesmos, mas também adquiridas de forma passiva através de observações não intrusivas, utilizando métodos implícitos.

Os métodos implícitos inferem informações dos usuários através do monitoramento do comportamento durante a interação com o sistema. Estes métodos

compreendem, geralmente, a coleta dos dados do tipo navegacionais e transacionais, através das seguintes verificações: histórico da navegação do usuário; transações efetuadas; páginas impressas, salvas ou adicionadas à lista de favoritos; e registro das palavras-chave utilizadas em sistemas de busca.

Os métodos implícitos e explícitos são complementares, permitindo obter um conjunto completo de informações sobre o usuário [Wasfí, 1999]. Além disso, a integração deles leva a sistemas menos intrusivos, considerando que os usuários não são requisitados a prover informações sobre suas preferências, características e interesses, mas ativamente participam na definição de seus modelos. Em trabalhos, tais como o de Joachims et al (1997) e Chittaro and Ranon (2002), os métodos explícitos são utilizados para compor um modelo inicial do usuário e os implícitos aplicados para a atualização do modelo.

Além disso, cabe destacar que os sistemas de recomendação, particularmente, adotam duas abordagens [Balabanovic and Shoham 1997; Adomavicius and Tuzhilin, 2001]: baseada em conteúdo e filtragem colaborativa.

Na abordagem baseada em conteúdo, o sistema considera as preferências individuais do usuário em relação a certas características dos objetos (produtos e páginas, por exemplo). Segundo Schwab and Kobsa (2002), as recomendações podem ser feitas considerando as características dos objetos selecionados anteriormente pelo usuário. Tal abordagem considera um modelo individual do usuário.

Na filtragem colaborativa, ou método de recomendação social conforme Basu et al (1998), a recomendação de um objeto é realizada a partir das escolhas de outros usuários que possuem características similares. Nessa abordagem, os usuários com interesses comuns são agrupados, permitindo a composição de modelos de grupo. O *Amazon.com* foi um dos primeiros sites comerciais a gerar recomendações personalizadas de produtos, com base em uma filtragem colaborativa [Levy and Weld, 2000].

Essa abordagem teve origem no trabalho de Rich (1983, 1989), no qual é proposta a organização de usuários com comportamentos comuns, em grupos (estereótipos). Neste processo, os grupos de usuários e as características típicas dos membros destes grupos são definidos na fase de desenvolvimento do sistema. Em tempo

de execução, os usuários são assinalados para um ou mais grupos e suas características são atribuídas conforme o grupo aos quais foram designados. Segundo Schwab and Kobsa (2002), a necessidade de uma pré-definição dos estereótipos é uma desvantagem no uso desta técnica. Como alternativa, Orwant (1995) e Paliouras et al (1999) propõem o uso de mecanismos de clusterização, onde grupos de usuários são dinamicamente encontrados, a partir dos modelos de usuários individuais que estão disponíveis. Conforme Paliouras et al (1999a), um estereótipo possui características comuns entre os usuários enquanto uma comunidade compartilha interesses comuns.

2.3 Técnicas de IA na Modelagem de Usuários

Diversas técnicas de IA têm sido aplicadas no processo de modelagem de usuários, provendo mecanismos automáticos e inteligentes para a aquisição de dados do usuário e indução de modelos. Dentre estas técnicas, pode-se destacar os métodos de aprendizado de máquina clássicos, as técnicas de reconhecimento de planos e as técnicas de mineração de dados. Nessa seção será dada ênfase às técnicas de aprendizado de máquina.

As técnicas de aprendizado de máquina visam dotar o sistema com um comportamento inteligente, provendo auxílio à tomada de decisões futuras, a partir do aprendizado de experiências passadas. Elas vêm sendo adotadas no processo de modelagem de usuários, principalmente, para adquirir modelos de usuários interagindo com o sistema. Em tais situações, os dados coletados das observações do comportamento do usuário são utilizados para construir um modelo de usuário individual [Papatheodorou, 2001; Paliouras et al 1999a] ou para prever o comportamento de um usuário como uma função dos dados de outros usuários [Levy and Weld, 2000].

Conforme Webb et al (2001) o uso de técnicas de aprendizado de máquina são adotadas, especialmente, na detecção e indução de padrões comportamentais ou preferências do usuário. Segundo esses autores, situações nas quais o usuário repetidamente realiza uma tarefa que envolve a seleção entre opções predefinidas parece ser ideal para a aplicação dessas técnicas para a formação de um modelo de usuário.

Para Schwab and Kobsa (2002), uma das formas de aplicação dos métodos de aprendizado de máquina na aquisição de modelos de usuários é assumir que

informações e objetos podem ser divididos em classes. As informações sobre os interesses (e desinteresses) dos usuários em cada classe são submetidas a um algoritmo de aprendizado, apto a distinguir objetos ou informações que podem interessar ou não ao usuário. O objetivo da utilização de um algoritmo de aprendizado é tornar automática a distinção entre o que é e o que não é interessante para o usuário, com base em experiências passadas (treinamento).

A aplicação de uma técnica de aprendizagem consiste nas etapas de aquisição dos dados do usuário (implícita ou explicitamente) e de fases treinamento e teste. Durante o treinamento, os dados coletados (exemplos para o treinamento) são submetidos ao algoritmo de aprendizado, e um modelo que representa os dados é “aprendido”. A fase de teste é aplicada para a avaliação do desempenho do algoritmo, a partir de um conjunto de exemplos de teste, utilizando um conjunto diferente do usado na fase de treinamento. A partir dessas etapas, o modelo “aprendido” é usado para representar os interesses ou prever ações futuras dos usuários.

A seleção da técnica de aprendizado depende largamente dos dados de treinamento disponíveis. Existem duas principais distinções entre estas técnicas: supervisionada e não supervisionada. As técnicas supervisionadas requerem que os dados de treinamento estejam previamente classificados (cada exemplo de treinamento é assinalado para uma classe). Esse processo é comumente referenciado como categorização ou classificação. Em contraste, as técnicas não supervisionadas não requerem a pré-classificação dos dados, agrupando os exemplos de treinamento a partir das características comuns que apresentam. Nessa técnica, cada exemplo de treinamento é assinalado para um *cluster*¹, processo referenciado como clusterização e adotado na formação de modelos de grupos de usuários.

Dentre os algoritmos de aprendizado de máquina que têm sido utilizados, destacam-se: as árvores de decisão, as redes neurais artificiais, o algoritmo *nearest neighbor* e os classificadores bayesianos.

As Árvores de Decisão (ADs) estão baseadas na utilização de algoritmos de particionamento recursivo, tais como o ID3 [Quinlan, 1986] e o C4.5 [Quinlan, 1993]. Nesses algoritmos, o conjunto de exemplos de treinamento é usado para induzir a

¹ Aglomerado de dados com alguma característica ou propriedade comum.

árvore, a partir da divisão recursiva dos exemplos em subconjuntos, utilizando a métrica de ganho de informação². Na árvore induzida, os nós internos correspondem aos atributos que compõem os exemplos; os ramos representam valores predeterminados para estes atributos; e os nós folha representam a categoria a qual o exemplo pertence. Após a construção da árvore, ela é usada na classificação de novos exemplos, descritos em termos dos mesmos atributos usados na sua representação. Isto é feito percorrendo-se a árvore aplicando o teste associado a cada nó (verificação do valor do atributo), até se chegar ao nodo folha, o qual determina a classe a que o exemplo provavelmente pertence.

As Redes Neurais Artificiais (RNAs) [Haykin, 2001] são sistemas compostos por unidades de processamento que simulam o comportamento de um neurônio biológico. Possuem a capacidade de aprender a partir dos exemplos de treinamento, e classificar ou clusterizar novos exemplos, com base na generalização do que aprenderam. São compostas por uma camada de entrada, formada por unidades que representam os dados correspondentes aos atributos utilizados para representar um exemplo; uma ou mais camadas intermediárias, formadas por neurônios responsáveis pela modelagem de relações entre as unidades de entrada e saída; uma camada de saída, que fornece a resposta do sistema (por exemplo, a probabilidade de um usuário pertencer a uma categoria, dados os valores de seus atributos); e ligações entre as várias camadas (pesos ou conexões sinápticas), responsáveis pela propagação de sinais entre elas e que representam o conhecimento adquirido pela rede durante o treinamento.

O algoritmo *nearest neighbor* (A-NN) [Mitchell, 1997], está baseado em um mecanismo de aprendizado que consiste em relembrar experiências anteriores (exemplos de treinamento). Esse algoritmo opera armazenando todos os exemplos disponíveis em um conjunto de treinamento, sendo cada exemplo representado como um ponto em um espaço pré-definido (espaço Euclidiano, por exemplo). A classificação de um novo exemplo é feita a partir da verificação do vizinho mais próximo no conjunto de treinamento (exemplo mais similar), utilizando uma medida de similaridade (distância entre eles, por exemplo). Desta forma, um novo exemplo é assinalado para a

² O ganho de informação é uma medida que indica a redução esperada na entropia de um conjunto de dados, causada pelo particionamento dos exemplos em relação a um dado atributo. Para obtenção do ganho de informação, calcula-se, inicialmente, a entropia: medida que indica a homogeneidade dos exemplos contidos em um conjunto de dados e permite caracterizar a “pureza” (e impureza) de uma coleção arbitrária de exemplos.

classe correspondente ao vizinho mais similar. Uma generalização para este algoritmo é o *k-nearest neighbor*, o qual considera não apenas um vizinho mais próximo, mas os *k* mais próximos para conduzir a classificação.

Um classificador bayesiano [Duda and Hart, 1973] é um método probabilístico aplicado à tarefa de classificação. Esses classificadores aprendem a partir dos exemplos de treinamento e geram uma estrutura de representação, com base no cálculo de probabilidades condicionais³. São utilizados para determinar a probabilidade de um novo exemplo pertencer a uma dada classe, considerando os valores de seus atributos, utilizando as probabilidades estabelecidas para o conjunto de treinamento. Dentre os tipos de classificadores bayesianos, os mais utilizados são os *naïve bayes*, os quais assumem que a probabilidade de ocorrência de uma conjunção de atributos em um dado exemplo é igual ao produto das probabilidades de ocorrência de cada atributo isoladamente (independência condicional dos atributos).

Entretanto, a modelagem de usuário apresenta características que dificultam a aplicação de técnicas de aprendizado de máquina. Webb et al (2001), apresenta os seguintes desafios: necessidade de grandes conjuntos de dados, necessidade de dados classificados, atualização do modelo e complexidade computacional.

Segundo Paliouras et al (1999a), um grande conjunto de dados de treinamento provê modelos de usuários mais robustos. Deste modo, a exatidão de um modelo de usuário é dependente da quantidade de dados disponíveis para treinamento do algoritmo de aprendizado. Dentre as possibilidades para amenizar o problema, Webb et al (2001) citam: uso de um modelo inicial do usuário, a partir do qual novos modelos são derivados; e utilização do algoritmo A-NN que apresenta uma performance aceitável, sem a utilização de grandes quantidades de exemplos de treinamento, classificando novos exemplos a partir da similaridade que apresentam com os exemplos de treinamento disponíveis.

Outro ponto a ser considerado diz respeito à utilização de métodos de aprendizado supervisionados, os quais requerem a classificações explícita e prévia dos exemplos de treinamento. Deste modo, é necessária a definição de classes e propriedades que descrevem apropriadamente cada classe [Abbattista et al, 2002]. No

³ Refere-se a probabilidade $P(C_i|A_1 = V_{1j} \dots A_n = V_{nj})$ de que um exemplo *j* pertence a uma classe *C*, dado que ele possui valores *V* para seus atributos *A*.

trabalho de Webb et al (2001) são descritas algumas soluções para o problema: explicitação de classes pelo usuário [Pazzani and Billsus, 1997; Schwab and Kobsa, 2002]; inferência de classes a partir de observações implícitas do comportamento do usuário [Lieberman, 1995]; e utilização de um conjunto inicial de exemplos classificados e aquisição de novos exemplos, similares aos do conjunto inicial (analisando documentos similares aos retornados por uma consulta realizada pelo usuário, por exemplo).

Outro desafio está na atualização do modelo. Segundo Billsus and Pazzani (1999) um modelo de usuário deve ser flexível para se adaptar as mudanças de interesses do usuário. Deste modo, os algoritmos de aprendizagem devem suportar a atualização dos modelos. Dentre as propostas para a atualização, pode-se citar: aplicação da técnica de janelamento (limitando os dados de treinamento conforme as n observações mais recentes do comportamento do usuário); e utilização de modelos duais [Chiu and Webb, 1999] – se um modelo treinado com dados recentes não oferecer exatidão suficiente, delega-se a classificação para um modelo treinado com dados anteriores.

Além disso, a aplicação de técnicas de aprendizado de máquina no contexto da Web, dadas a quantidade de informações e o número de usuários, pode ser inviabilizada devido ao alto custo computacional. Assim, a utilização de algoritmos considerados computacionalmente caros ou que realizam a análise de extensas quantidades de dados devem ser aplicados em cenários onde os modelos podem ser aprendidos *offline*.

Por fim, Abbattista et al (2002) e Schein et al (2002) citam o problema, conhecido como *cold-start problem*, especialmente experimentado em sistemas de recomendação, onde recomendações são requeridas para novos usuários e pouca ou nenhuma informação está disponível para predizer as preferências do mesmo. De fato, para estar apto a produzir predições exatas, o sistema deve, primeiro, coletar informações suficientes sobre o usuário. Para minimizar o problema, poderia ser utilizada a abordagem apresentada por Pazzani and Billsus (1997), que utiliza conhecimento léxico para auxiliar na elaboração de um modelo inicial a partir da análise de palavras relacionadas. Em uma abordagem recente, Middleton et al (2002) exploram o uso de ontologias para investigar como o domínio do conhecimento pode auxiliar na

aquisição de preferências do usuário, provendo conhecimento inicial sobre os usuários e seus domínios de interesse.

É pertinente salientar que, além das técnicas de aprendizado de máquina clássicas, outras técnicas de IA têm sido adotadas em modelagem de usuários. As técnicas de reconhecimento de planos [Bauer, 1996; Carberry, 2001] inferem objetivos do usuário através do reconhecimento dos planos usados para atingir estes objetivos. A partir das observações do comportamento do usuário, são identificadas as seqüências de ações, determinados os planos e previstas as próximas ações. Além disso, as técnicas de mineração de dados vêm sendo utilizadas [Adomavicius and Tuzhilin, 2001]. Tais técnicas objetivam descobrir informações implícitas em dados explícitos e têm sido aplicadas, especialmente, para a descoberta de padrões de uso, a partir da observação do comportamento do usuário, permitindo a inferência de modelos.

2.4 Trabalhos Relacionados

A importância de adicionar a capacidade de adaptação aos sistemas, conforme conhecimento sobre os seus usuários, é comprovada pela variedade de aplicações na qual a modelagem de usuários vem sendo empregada: sistemas de recuperação e filtro de informações, comércio eletrônico, softwares educacionais e hipermídia adaptativas.

Em sistemas de recuperação e filtro de informações o uso de um modelo de usuário objetiva disponibilizar aos usuários informações relevantes aos seus interesses e filtrar as irrelevantes. Nas aplicações de comércio eletrônico, a personalização tem se tornado uma importante estratégia de negócio. Segundo Kobsa (2001), a personalização permite migrar o relacionamento com consumidores na Internet do marketing de massa anônimo para o marketing *one-to-one*. Em softwares educacionais, um modelo de usuário é utilizado para adaptar o ambiente conforme as habilidades e conhecimento do aluno. Além disso, os sistemas de hipermídia adaptativa [Brusilovsky, 2001] têm sido vistos como a união da hipermídia com a modelagem de usuários. Tais sistemas produzem um modelo dos objetivos, preferências e conhecimento de cada usuário, utilizando-o para adaptar o conteúdo de uma página ou sugerir relevantes a seguir.

Nesta seção, sistemas que fazem uso de um modelo de usuário são comentados.

Um dos primeiros sistemas a adotar um modelo de usuário para recomendação de páginas foi o Letizia [Lieberman, 1995]. Letizia é um agente que observa o comportamento do usuário durante a navegação pela web e registra evidências positivas e negativas de interesses. Por exemplo, visitar uma página indica interesse na página, adicioná-la aos favoritos, indica forte interesse, enquanto passar sobre um link, sem visitá-lo, é visto como um desinteresse. A partir destas evidências, são feitas análises dos conteúdos das páginas, utilizando o algoritmo TF-IDF⁴. Desta análise, resulta uma lista de palavras, com pesos correspondentes, utilizadas para compor o modelo do usuário. O modelo é utilizado, então, para a recomendação de hiperlinks a partir da página corrente do usuário.

Um sistema similar é o Personal WebWatcher, proposto por Joachims et al (1997). A recomendação de páginas é feita comparando-as com as páginas anteriormente visitadas pelo usuário. O sistema gera um modelo de usuário, composto pelo conjunto das palavras presentes em cada página visitada (aplicando o algoritmo TF-IDF), e sugere hiperlinks, comparando o conteúdo destas páginas com o modelo do usuário. O método de aprendizado por reforço é agregado ao processo de recomendação, como forma de melhorar a exatidão da recomendação.

O sistema Syskill & Webert, apresentado por Pazzani and Billsus (1997), é outro exemplo de sistema de recomendação de páginas. Um conjunto de documentos pré-selecionados é avaliado (interessante ou não interessante) pelo usuário. Baseados nestas avaliações, o sistema identifica palavras informativas de cada página (utilizando a métrica de ganho de informação) e utiliza-as como características booleanas no aprendizado de um classificador bayesiano. O classificador é utilizado para recomendar novas páginas ao usuário, a partir da computação da probabilidade de uma página ser interessante ou não interessante, dado o seu conjunto de palavras.

O trabalho de Perkowitx and Etzioni (1998,1999) foi um dos primeiros a apresentar a idéia de reestruturação automática da organização e apresentação de um web site, conforme o aprendizado de padrões de uso do mesmo. Os dados coletados a partir da observação da navegação dos usuários pelo site são submetidos a um algoritmo

⁴ O algoritmo TF-IDF (*term frequency – inverse document frequency*) assinala para cada termo em um documento, um valor numérico que indica a representatividade do termo para o documento. O valor é computado dividindo a frequência do termo no documento pela frequência total do termo, em todos os documentos.

de clusterização, através do qual são detectados conjuntos de páginas relacionadas. Os clusters detectados são, então, utilizados para a indexação de conjuntos de páginas do site, de forma a facilitar a navegação do usuário pelo site.

Um agente inteligente, chamado NewsDude, designado a aprender os interesses dos usuários em páginas de notícias, a partir da observação das ações e feedback explícito dos usuários, é proposto por Billsus and Pazzani (1999). Uma abordagem multi-estratégia é utilizada para manter um modelo do usuário, refletindo os interesses de curto e longo prazo deste. Os interesses de curto prazo são coletados a partir das n mais recentes observações, adotando um algoritmo *nearest neighbor* para representar estes interesses. Um classificador bayesiano é adotado para aprender os interesses de longo prazo. Para a recomendação de novas notícias, estas são submetidas ao modelo de curto prazo. Caso não possam ser classificadas (dado um limiar de similaridade entre as representações do modelo e da notícia), são submetidas ao modelo de longo-prazo.

Adomavicius and Tuzhilin (2001) propõem a aplicação de técnicas de mineração de dados para descobrir um conjunto de regras que descrevem o comportamento dos usuários de um site de comércio eletrônico, a partir dos dados transacionais coletados. Um modelo do usuário é construído a partir das regras encontradas e utilizado para a personalização do site.

A modelagem de estudantes, em um sistema de hipermídia educativa, utilizando RNAs, é proposta por Castellano et al (2001). As características dos estudantes são agrupadas na forma de clusters, encontrados a partir da utilização de uma rede neural competitiva. Um questionário é apresentado ao estudante e as respostas são submetidas à rede neural, responsável pela geração dos clusters. Cada cluster é utilizado para representar um modelo de grupo de estudantes.

Abbattista et al (2002) propõem a extração de modelos de usuários a partir da análise de arquivos de log de acesso, em um site especializado em livros. É utilizada uma técnica de aprendizado de máquina supervisionada, para induzir um classificador apto a discriminar entre itens interessantes e não interessantes ao usuário – da análise dos arquivos são extraídas as características usadas como exemplos (previamente classificados por um especialista humano) para o classificador, aplicado na recomendação de novos itens. Como forma de aumentar a exatidão da recomendação, é

aplicado o processo de categorização de textos. É feita a coleta de descrições de livros de diferentes categorias, e os mesmos são avaliados pelos usuários (interessante ou não interessante). Tais avaliações compõem os exemplos de treinamento de um classificador bayesiano, apto a discriminar entre livros interessantes e não interessantes para cada categoria preferida pelo usuário. Além disso, é adotado um algoritmo de clusterização para inferir padrões de uso de grupos de usuários.

Muitos trabalhos utilizam algoritmos de aprendizado que consideram evidências positivas e negativas acerca dos interesses dos usuários. Schwab and Kobsa (2002) propõem uma abordagem para o aprendizado de um modelo de usuário utilizando apenas evidências positivas, coletadas implicitamente (observações do comportamento do usuário). Tal abordagem elimina a necessidade de solicitar aos usuários avaliações explícitas ou evidências negativas, indicando seus desinteresses. Com base nos objetos selecionados pelo usuário (considerados interessantes) são extraídas as características utilizadas para compor o modelo. A recomendação de novos objetos é feita utilizando a abordagem baseada em conteúdo, utilizando um o algoritmo *k-nearest neighbor* para aprender uma caracterização dos objetos de interesse, e a colaborativa, recomendando objetos que usuários similares selecionaram no passado.

Conforme visto, os esforços na utilização de um modelo de usuário estão concentrados em interfaces 2D. O trabalho de Chittaro and Ranon (2000, 2000a, 2002, 2002a) é o pioneiro na adoção de um modelo de usuário para a personalização de um ambiente 3D. A proposta destes autores é utilizar um modelo do usuário na adaptação da estrutura e *layout* de uma loja virtual. Um modelo inicial é coletado a partir da aplicação de formulários, onde são requisitadas informações típicas (tais como gênero, idade, profissão e nível de escolaridade) e específicas (categorias de produtos de interesse, preferência do tamanho e estilo da loja e presença ou ausência de música na loja). Caso o usuário não informe todas os dados solicitados, são utilizados estereótipos para produzir previsões sobre os interesses e preferências. A partir do modelo inicial é estabelecido um ranking com os produtos de interesse. A atualização do modelo inicial é realizada a partir da análise do comportamento do usuário (seções da loja visitadas e produtos comprados) e com base em regras, como por exemplo: $Se\ visitou(prodoto) = 0\ e\ numeroVisitas > 3\ entao\ decrementaInteresse(prodoto)$. Com a atualização do modelo, um novo ranking dos produtos de interesse é obtido.

Conforme comentado neste capítulo, diversas abordagens e técnicas para a modelagem de usuários vêm sendo propostas e, como consequência disto, aplicações que colecionam diferentes tipos de informações sobre os seus usuários e exibem diferentes formas de adaptação têm sido desenvolvidas. No entanto, pode-se destacar os requisitos fundamentais associados a modelagem de usuários. Um modelo de usuário deve ser formado por dados coletados de forma explícita e implícita. As indicações explícitas dos usuários sobre os seus interesses e preferências, especialmente nas primeiras interações, evita o problema *cold-start*. Por outro lado, as hipóteses de interesse devem também ser adquiridas de forma passiva, ao longo da interação com o sistema, permitindo que o modelo seja flexível e adaptável aos novos interesses dos usuários, refletindo os interesses de curto e longo prazo dos mesmos. Neste contexto, é fundamental considerar a incerteza associada as hipóteses coletadas implicitamente, de forma que possam ser quantificados os graus de interesse dos usuários, e a utilização de uma janela de tempo customizável, permitindo que sejam consideradas as n observações mais recentes do comportamento do usuário. Além disso, a aplicação de técnicas que automatizem o processo de aquisição e atualização dos modelos, a um custo computacional baixo se faz necessária. Por fim, cabe destacar a importância do processo de análise dos conteúdos (páginas *web*, por exemplo) para a composição de modelos de usuários. Diversos autores propõem a aplicação de técnicas de aprendizado de máquina na categorização de conteúdos, como uma tarefa associada à modelagem de usuários. Nesta dissertação, é apresentada a modelagem de conteúdos como uma tarefa independente da modelagem de usuários. A partir da combinação de modelos de usuários e conteúdos, as adaptações no ambiente tridimensional são realizadas. O próximo capítulo apresenta o processo de modelagem de conteúdos, proposto nesta dissertação.

3 Modelagem de Conteúdos

A crescente quantidade de informação disponibilizada eletronicamente, principalmente através da Internet, e a dificuldade de recuperação das mesmas têm motivado o desenvolvimento de pesquisas na área de Recuperação de Informação (RI). O objetivo é auxiliar os usuários na localização de informações relevantes aos seus interesses.

Inúmeros sistemas de busca na Internet adotam as técnicas de recuperação baseadas em palavras-chave. Nestes sistemas, não é realizada uma análise mais detalhada dos conteúdos e muitas das informações retornadas aos usuários são de pouca ou nenhuma relevância. Isso se deve ao fato de que grande parte dos conteúdos é descrita apenas por um conjunto de palavras-chave, definido pelo especialista responsável pelos conteúdos. Sob essa perspectiva, uma das alternativas consiste na utilização de modelos de conteúdos, que podem ser utilizados para descrevê-los mais detalhadamente. A partir disso, os mecanismos de busca podem fazer uso dos modelos e refinar os resultados das consultas, auxiliando os usuários na recuperação de informações relevantes e filtragem das irrelevantes.

Uma abordagem que pode auxiliar na criação de modelos de conteúdos consiste na aplicação de um processo de categorização. Neste processo, é feita uma análise mais detalhada da informação que compõe os conteúdos, sendo os mesmos, então, organizados em categorias pré-definidas (por exemplo, esporte, economia, lazer, turismo, entre outras). Em sistemas, tais como o *Yahoo*⁵, a categorização é adotada, mas feita de forma manual. No entanto, categorizar grandes volumes de conteúdos manualmente tem se tornado inviável.

Atualmente, tem sido proposta a aplicação de técnicas de IA na automatização do processo de categorização. Abordagens utilizando técnicas de aprendizado de máquina, tais como Árvores de Decisão (ADs), Redes Neurais Artificiais (RNAs) e classificadores Bayesianos, vêm sendo exploradas. Nestas abordagens, as ferramentas

⁵<http://www.yahoo.com>.

são treinadas para oferecer suporte a problemas de decisão (tais como classificação ou clusterização), baseadas em dados de treinamento.

De um modo geral, o processo de categorização automático ou clusterização vem sendo adotado em conteúdos textuais [vide Sebastiani, 2002]. Neste caso, o processo pode ser utilizado na extração de características dos conteúdos e composição de um modelo de conteúdo. Dentre as aplicações que podem ser exploradas com esta abordagem pode-se citar a organização de mensagens de e-mail, o filtro de mensagens (evitando as do tipo *spam*, por exemplo) e notícias, a organização hierárquica e a recomendação de documentos.

Entretanto, cabe destacar as pesquisas relacionadas com a extração de características de conteúdos dos tipos imagem [vide Min et al, 2003; Bender and Osório, 2003] e vídeo [vide Agius and Angelides, 2001]. Neste caso, o objetivo é fornecer mecanismos para o reconhecimento e a recuperação de imagens (baseados em alguma característica, tais como cor, textura e/ou forma) e vídeos (baseados na identificação eventos e interações entre objetos, por exemplo).

No contexto desta dissertação é abordado o processo de categorização automático de conteúdos textuais, utilizado como auxiliar na aquisição de modelos de conteúdos. Estes modelos são utilizados na organização espacial dos conteúdos no ambiente tridimensional e, juntamente com os modelos de usuários, são aplicados no processo de adaptação do ambiente.

Este capítulo apresenta conceitos básicos relativos à modelagem de conteúdos e está organizado como segue. Na seção 3.1 são comentados os modelos de conteúdos. A seção 3.2 apresenta o processo de modelagem de conteúdos, como uma tarefa associada ao processo de categorização automático. Na seção 3.3, são apresentados os trabalhos que fazem uso de um processo de categorização de textos. A seção 3.4 apresenta a metodologia para o processo de modelagem proposto nesta dissertação e, por fim, na seção 3.5 são expostos os experimentos preliminares realizados com a aplicação de técnicas de aprendizado de máquina em categorização de textos e os resultados obtidos.

3.1 Modelo de Conteúdo

Um modelo de conteúdo pode ser visto como uma coleção de informações sobre um conteúdo, usada para descrevê-lo. Dentre os dados que podem ser utilizados na descrição, pode-se citar: título, autor, data, localização, descrição, categoria do conteúdo (área do conhecimento a que pertence), tipo de mídia e palavras-chave que o caracterizam. Ele pode ser visto como um metadado associado ao conteúdo, que provê uma descrição estruturada do conteúdo, facilitando a catalogação e a recuperação do mesmo.

Dentre os padrões de metadados que vem sendo aplicados na descrição de recursos, destaca-se o *Learning Object Metadata*⁶ (LOM), investigado pelo *Institute of Electrical and Electronics Engineers* (IEEE). O LOM especifica a sintaxe e a semântica para a descrição de recursos educacionais, digitais ou não digitais, tais como conteúdos multimídia, conteúdos instrucionais, softwares educacionais, pessoas, organizações e eventos. O propósito do padrão é definir o conjunto de atributos necessário para que estes recursos possam ser facilmente gerenciados, localizados, compartilhados e avaliados. Apesar do padrão objetivar a descrição de recursos educacionais, certo conjunto dos atributos poderia ser aplicado na descrição de recursos que não são necessariamente educacionais, tais como documentos em geral, vídeos e imagens, formando modelos destes recursos.

Para esta dissertação, objetiva-se a aquisição automática das principais características que podem ser utilizadas na composição de modelos de conteúdos textuais (categoria do conteúdo e conjunto de palavras-chave correspondente), a partir de um processo de categorização. Outras características que podem ser consideradas na formação do modelo (autor e tipo de mídia, por exemplo) são definidas pelo especialista responsável pelo conteúdo. Além disso, para conteúdos não textuais, o especialista atua diretamente na composição do modelo. Neste contexto, considerando o LOM como um padrão aplicável à descrição de conteúdos, uma perspectiva seria prever a geração dos modelos de conteúdos seguindo o padrão, facilitando o gerenciamento e compartilhamento dos conteúdos.

⁶ <http://ltsc.ieee.org/wg12/>

3.2 Processo de Modelagem

O processo de modelagem de conteúdos envolve a coleta das informações necessárias à elaboração do modelo e a representação das informações coletadas. Este processo, independente das características utilizadas para descrever o conteúdo, geralmente, é feito de forma manual. Entretanto, especialmente para os conteúdos textuais, pode ser adotada uma categorização automática de textos, a partir da qual podem ser extraídas algumas características, tais como a categoria do conteúdo e o conjunto de palavras-chave utilizado para caracterizá-lo. Nesta seção, será abordado o processo de categorização automática de conteúdos textuais.

Conforme Lewis (1991) e Yang and Liu (1999) a categorização é uma técnica empregada para identificar a categoria a que determinado documento pertence, utilizando como base o seu conteúdo. Deste modo, as categorias devem ser previamente descritas através de suas características [Rijsbergen, 1979].

O processo de categorização, independente da ferramenta de aprendizado adotada, é formado por um conjunto de etapas fundamentais. A Figura 3.1 ilustra o processo de categorização.

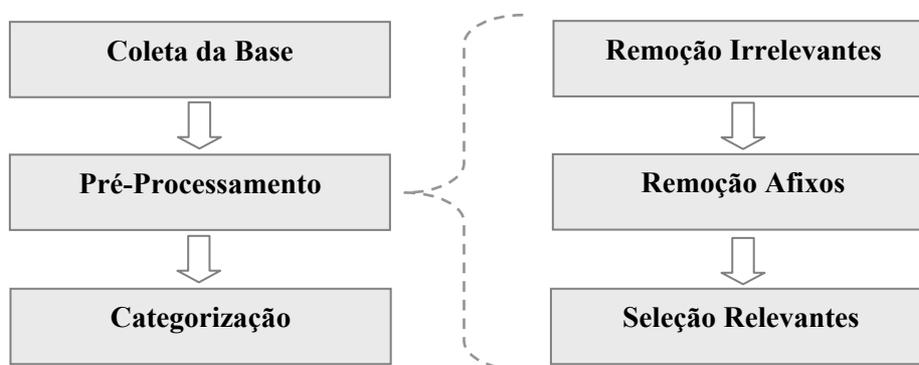


Figura 3.1 – Processo de categorização.

A etapa de coleta da base consiste na obtenção dos exemplos a serem utilizados para o aprendizado e a validação do categorizador. Nesta etapa, são definidos o tipo de categorizador adotado e as categorias e o conjunto de exemplos correspondentes. Dois tipos de categorizadores podem ser adotados: binário ou múltiplo. Com os categorizadores binários, um exemplo é categorizado como pertencente ou não a uma determinada categoria. Nos categorizadores múltiplos, um exemplo é atribuído a uma

categoria, dentre um conjunto de n categorias. Por exemplo, considerando três categorias, com o uso do categorizador binário, cada categoria tem um categorizador associado e, deste modo, um exemplo pode estar associado a mais de uma categoria; com um categorizador múltiplo, um exemplo é atribuído para uma categoria, dentre as três pré-definidas.

Após a coleta dos exemplos, é feito o pré-processamento de cada exemplo. Nesta etapa, são eliminadas as palavras consideradas irrelevantes (*stopwords*, tais como artigos, preposições, pronomes e caracteres especiais), é realizada a remoção dos afixos das palavras (prefixos e sufixos) e é feita a seleção das palavras relevantes utilizadas para caracterizar o documento correspondente. A remoção dos afixos é necessária para que palavras, como por exemplo *neural* ou *neurais*, sejam consideradas unicamente.

Quanto aos métodos de seleção de palavras relevantes, podem ser utilizados, entre outros, os seguintes: seleção baseada no peso do termo (palavra), seleção por indexação semântica latente, seleção por linguagem natural e seleção baseada na teoria da informação.

A seleção *baseada no peso do termo* [Yang and Pedersen, 1997], comumente utilizada, consiste em filtrar os termos cujo peso respeite um limiar (*threshold*) pré-estabelecido. O peso de um termo é um valor numérico que especifica a representatividade do mesmo para o documento ao qual pertence. Para o cálculo do peso são geralmente adotados os métodos baseados em frequências, tais como frequência absoluta⁷, frequência relativa⁸ e frequência do termo - frequência inversa de documentos⁹. Considerando que a aplicação desta técnica pode levar à seleção de uma grande quantidade de termos (por exemplo, seleção dos termos com frequência absoluta > 5), pode ser adotado um método de seleção que estabelece o número máximo de termos que podem ser selecionados. Deste modo, a partir da ordenação dos termos, de acordo com os pesos correspondentes, é possível selecionar apenas os n primeiros termos para caracterizar um documento (por exemplo, seleção dos 20 primeiros termos que ocorrem nos documentos, tendo os mesmos ordenados por frequência absoluta).

⁷ Número de vezes que a palavra ocorre no documento.

⁸ Número de vezes que a palavra ocorre no documento, dividido pelo número de palavras no documento.

⁹ Peso é computado dividindo a frequência da palavra no documento pela frequência total da palavra, em todos os documentos. O método é conhecido como TF-IDF (*term frequency – inverse document frequency*).

Outro método para a seleção de palavras relevantes é o de *indexação semântica latente* [Wiener et al, 1995], baseado na aplicação de uma análise da estrutura correlacional de termos nos documentos, visando extrair associações entre os termos e reduzir o número de termos selecionados. Além disso, seleção por linguagem natural [Salton and MacGill, 1983], agrega as análises sintática e semântica dos documentos na extração dos termos relevantes (por exemplo, podem ser considerados apenas os substantivos ou os adjetivos no processo de extração dos termos relevantes). Por fim, a *seleção baseada na teoria da informação* utiliza métricas, tais como o ganho da informação [Yang and Pedersen, 1997], coeficiente de correlação [Ng et al, 1997] e qui-quadrado [Ng et al, 1997], na definição dos termos relevantes.

Após realizada a seleção das palavras relevantes, utilizando um dos métodos citados acima, a próxima etapa do processo consiste na categorização. Nesta etapa, é determinada a técnica de aprendizado e é feita a codificação de cada exemplo. A codificação geralmente adota uma representação vetorial, onde as palavras selecionadas como relevantes tornam-se índices de vetores e os respectivos valores correspondem ao peso da palavra no documento correspondente. Cabe destacar que na codificação, geralmente, é adotado o peso baseado no cálculo da frequência ou o peso binário (presença ou ausência da palavra no documento). Além disso, ainda na etapa de categorização são definidos os parâmetros de aprendizado e os conjuntos de exemplos de aprendizado e validação, a serem submetidos à ferramenta de categorização, e é realizada a categorização propriamente dita.

Após estas etapas, o categorizador pode ser utilizado na categorização de novos documentos.

3.3 Trabalhos Relacionados

Conforme comentado, diversas abordagens têm sido propostas para a categorização automática de documentos com a aplicação das ferramentas de aprendizado de máquina. Nesta seção, serão brevemente comentados os trabalhos relacionados.

Pazzani and Billsus (1997) apresentam a categorização de páginas *web* como uma tarefa associada a modelagem de usuários. Na abordagem proposta, comentada na seção 2.4, um conjunto de páginas avaliadas pelo usuário é utilizado como entrada para um classificador bayesiano, apto a identificar novas páginas que são interessantes (ou

desinteressantes) para o usuário. Abordagens similares são utilizadas por Lieberman (1995), Joachims et al (1997), e Billsus and Pazzani (1999).

De outro modo, a aplicação do processo de categorização como uma tarefa relacionada a organização de documentos é proposta por diversos autores. Oliveira and Castro (2000) propõem a aplicação de ADs para a categorização múltipla de textos. O processo de aprendizado é efetuado sobre todas as categorias dos textos, gerando um único classificador capaz de associar cada novo documento a uma ou mais categorias pré-definidas. São usadas para os experimentos as coleções *Reuters – 21578* (um conjunto de 21.578 artigos da agência de notícias *Reuters*, devidamente classificados) e *Medline* (formada por artigos da área médica).

A categorização de textos na organização e filtragem de informações na área empresarial, utilizando RNAs, é proposta por Rizzi et al (2000). O objetivo é validar a aplicação do processo na atividade empresarial, especialmente nas tarefas de análise e distribuição das informações. Um conjunto de experimentos, utilizando a coleção *Reuters – 21578* e a RNA *Multi-layer – Perceptron* (MLP), com o algoritmo de aprendizado *Backpropagation* (BP), é apresentado.

Melcop et al (2002) adotam as ADs na classificação de páginas *web*. É apresentado um sistema para recuperação e classificação de páginas de domínios específicos, a partir da utilização de uma ontologia do domínio escolhido. No sistema de classificação, as árvores geradas são capazes de definir se um determinado documento pertence ou não a uma dada categoria (categorização binária). Experimentos para o domínio da culinária são apresentados.

Correa and Ludemir (2002) apresentam e comparam o uso das RNAs, dos tipos MLP (com algoritmo de aprendizado BP) e *Self-organizing Maps* (SOM) e das técnicas tradicionalmente aplicadas no processo de categorização, tais como ADs e classificadores *Naive Bayes*. Foram utilizadas nos experimentos as coleções de dados *K1* (uma coleção de 2.340 páginas *web* sobre notícias diversas), *PubsFinder* (um conjunto de 1.300 páginas *web* classificadas como sendo ou não páginas de publicação de artigos técnicos e científicos) e a sub-coleção *Metals* da *Reuters – 21578*.

Duarte et al (2002) propõem o uso de um agente neural para a coleta e classificação de informações disponíveis na Internet. O agente coleta as informações,

submetendo-as a um classificador neural que utiliza uma RNA do tipo MLP. Foram utilizados os algoritmos *Levenberg Marquadt* e *MOBJ* para o treinamento da rede. Experimentos para o domínio da economia são apresentados.

Na abordagem proposta por Silva et al (2003; 2003a), são aplicadas análises morfossintáticas na extração de palavras relevantes para os processos de categorização e clusterização. São apresentados experimentos com a coleção de artigos jornalísticos do NILC – UFSCar (Núcleo Interdisciplinar de Lingüística Computacional – Universidade Federal de São Carlos) e o uso das técnicas de aprendizado de máquina ADs e RNAs (com os algoritmos *CascadeCorrelation* e BP), para a categorização, e o algoritmo *k-means*¹⁰ para a clusterização.

Para esta dissertação, foi definida uma metodologia para a categorização de textos, aplicando as ADs e as RNAs MLP, com os algoritmos BP e *Cascade Correlation* (CasCor), e utilizando uma base de dados de pequeno porte e métodos de seleção baseados em frequência. O objetivo foi avaliar a performance dos algoritmos, de forma a selecionar aquele que apresentou menores taxas de erros, para uso posterior no processo de aquisição de modelos de conteúdos. Para atingir tal objetivo, um conjunto de experimentos foi realizado. Nas seções que seguem, a metodologia e os resultados obtidos com os experimentos são comentados.

3.4 Metodologia

Diferentemente dos trabalhos apresentados na seção 3.3, para o processo de categorização proposto nesta dissertação, foi utilizada uma base de dados de pequeno porte, objetivando verificar a performance e a praticidade das ferramentas de categorização. Dado que estas adotam algoritmos de aprendizado supervisionados, é exigida a classificação prévia de um conjunto de exemplos. Entretanto, em sistemas que adotam uma variedade considerável de categorias, é indispensável o uso de ferramentas que sejam customizáveis, sem que uma grande quantidade de exemplos de aprendizado se torne necessária na inserção de novas categorias, viabilizando o uso do processo.

¹⁰ Método de agrupamento (clusterização) baseado no particionamento de um conjunto de n objetos em k grupos (clusters) de modo que a similaridade intracluster resultante é alta, mas a similaridade intercluster é baixa [Theodoridas and Koutroumbas, 1998].

A metodologia consistiu na definição das categorias, coleta dos exemplos correspondentes e aplicação das técnicas simbólica (ADs) e conexionista (RNAs). Para os experimentos preliminares, foram adotados os categorizadores múltiplos e definidas as seguintes categorias: Redes Neurais Artificiais (RNAs), Algoritmos Genéticos (AGs) e Sistemas Multiagentes (SMAs). Para cada categoria, foram coletados 30 exemplos de artigos científicos (a partir dos mecanismos de busca da *web*), em língua inglesa, e extraídos os seus *abstracts*, introdução e conclusão.

Após a coleta da base de dados, os exemplos selecionados foram pré-processados por uma aplicação, que contempla as sub-etapas correspondentes (Figura 3.1) e gera os scripts (com os documentos devidamente codificados) que são submetidos às ferramentas de aprendizado.

A aplicação, implementada em Java, foi estendida de um *framelet*¹¹ (vide Pree and Koskimies, 1999) desenvolvido para esta dissertação. O *kernel* do *framelet* contempla o fluxo básico de dados entre as atividades de remoção das palavras irrelevantes e afixos, seleção das relevantes e geração dos scripts. A partir do preenchimento dos *hot – spots* do *framelet*, as atividades podem ser customizadas de acordo com as necessidades da aplicação. Assim, o *framelet* oferece a flexibilidade suficiente para a geração de diferentes formatos de arquivos de scripts, o que permitiu a realização de variados experimentos.

Na aplicação que estende o *framelet*, os *hot – spots* foram preenchidos conforme os métodos e técnicas comentadas a seguir. Para o processo de remoção das palavras irrelevantes foi utilizada a lista de *stopwords* para a língua inglesa elaborada pelo Laboratório de Recuperação de Informações da Universidade de Massachusetts¹². O algoritmo de extração de afixos adotado foi o proposto por Martin Porter¹³, que remove as letras do final de palavras da língua inglesa que possuem a mesma variação morfológica e de flexão. Para a seleção das palavras mais relevantes, foi utilizado o algoritmo de seleção por frequência, onde foram selecionadas as *n* palavras que mais ocorrem no conjunto de documentos de cada categoria.

¹¹ *Framelet* e aplicação disponíveis em <http://www.inf.unisinos.br/~cassiats/dissertacao>

¹² Disponível em: http://www.cs.umass.edu/Dienst/UI/2.0/Describe/ncstrl.umassa_cs%2FUM-CS-1991-093

¹³ Disponível em: <http://www.tartarus.org/~martin/PorterStemmer/index.html>

Após o pré-processamento, foram elaborados os vetores representativos de cada categoria – vetores locais – e da base de exemplos (união dos vetores locais) – vetores globais. Cabe destacar que os vetores locais foram elaborados a partir das n palavras mais freqüentes do conjunto de exemplos correspondentes. Em seguida, foi realizada a codificação de cada exemplo. As palavras nos vetores globais serviram como índices para os vetores de cada exemplo e as posições correspondentes representaram a importância da mesma no documento (dependendo do tipo de codificação utilizado).

Em relação aos tipos de codificação, a por freqüência consiste na utilização das freqüências das palavras na construção dos vetores, enquanto a binária considera a presença (1) ou ausência (0) da palavra. Após a codificação dos exemplos, foram gerados os scripts para as ferramentas de suporte aos algoritmos de aprendizado: C4.5 [Quinlan, 1993] e Neusim [Osorio and Amy, 1999]. O C4.5 é a ferramenta responsável pela geração das árvores de decisão, enquanto o Neusim é um simulador neural que comporta os algoritmos de aprendizado BP e CasCor. Na seção seguinte, os experimentos realizados com o uso das ferramentas são comentados.

A Figura 3.2 ilustra as etapas relacionadas ao pré-processamento e geração dos scripts, conforme a metodologia proposta.

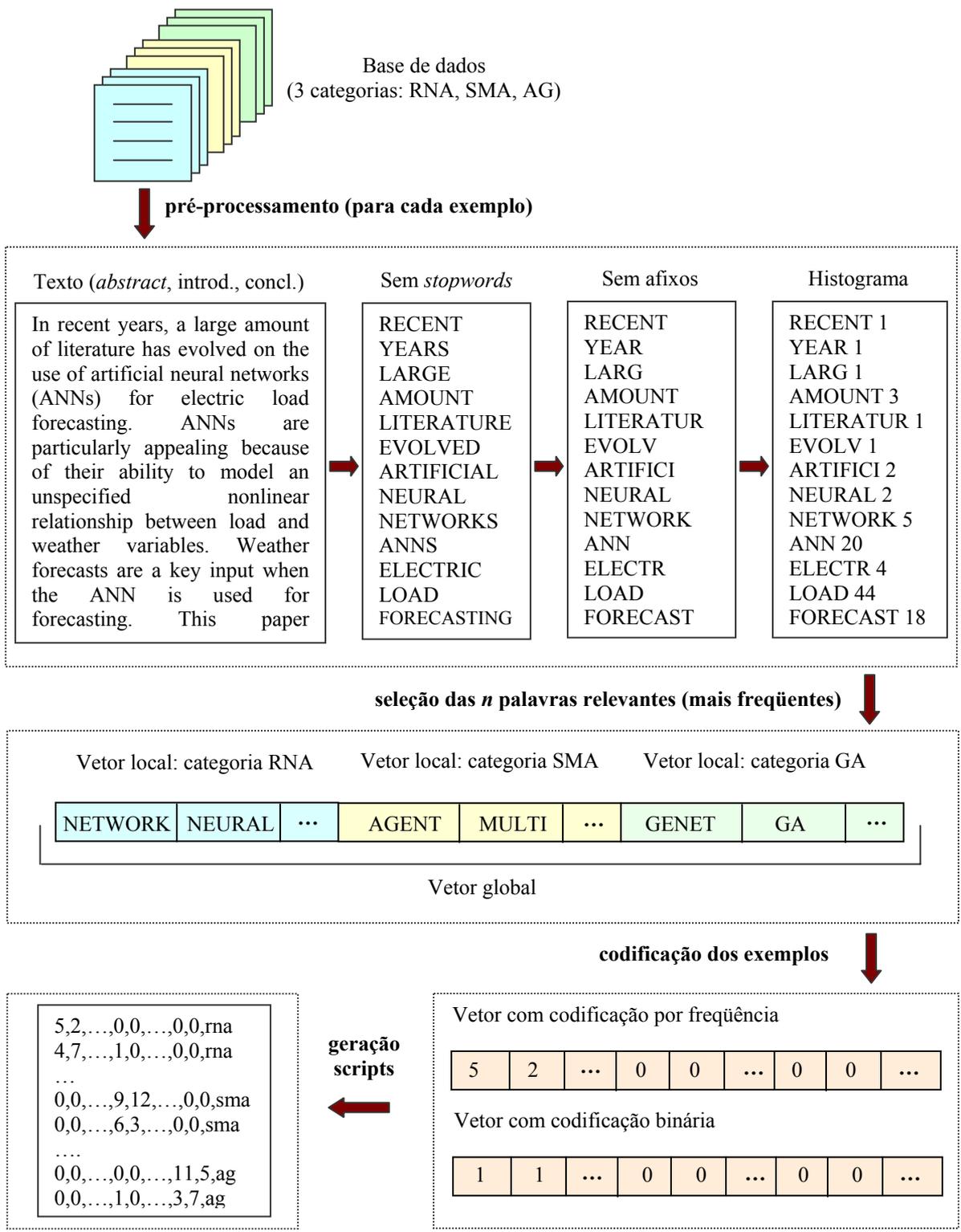


Figura 3.2 – Etapas de pré-processamento e geração dos scripts, conforme a metodologia proposta.

3.5 Experimentos Preliminares

Nos experimentos¹⁴ realizados foram consideradas variações nos seguintes parâmetros: tipo de categorização, tipo de codificação, número de palavras relevantes selecionadas por categoria e tipo de algoritmo de aprendizado. Em todos os experimentos foram utilizadas a categorização múltipla, as codificações binária e por frequência, e a seleção de 20, 50 e 80 palavras relevantes por categoria. A categorização múltipla foi adotada na geração dos scripts, onde os arquivos de aprendizado e validação contemplaram os exemplos das 3 categorias definidas. A base de dados utilizada contemplou um total de 90 exemplos, divididos em conjuntos de aprendizado (2/3 dos exemplos) e validação (1/3 dos exemplos).

Os primeiros experimentos foram feitos com as ADs e as Regras de Produção (RPs) extraídas das árvores, utilizando-se a ferramenta C4.5. As Tabelas 3.1 e 3.2 indicam os resultados percentuais obtidos na geração das árvores e regras, respectivamente, em relação ao erro de classificação, no conjunto de validação (erro na generalização). Indica-se, também, o número de nodos de cada árvore e o número de regras extraídas. Cabe ressaltar que, na seleção das 20, 50 e 80 palavras mais relevantes, o número de atributos do vetor global foi de 44, 105 e 161, respectivamente. Estes números foram obtidos porque as palavras repetidas nos vetores locais foram eliminadas na formação do vetor global.

Tabela 3.1 – Erro de classificação (ADs).

	No. Palavras por Categoria					
	20		50		80	
Codificação	<i>Erro</i>	<i>Nodos</i>	<i>Erro</i>	<i>Nodos</i>	<i>Erro</i>	<i>Nodos</i>
Binária	3.3%	7	3.3%	7	3.3%	7
Frequência	3.3%	7	3.3%	7	3.3%	7

Tabela 3.2 – Erro de classificação (RPs).

	No. Palavras por Categoria					
	20		50		80	
Codificação	<i>Erro</i>	<i>Regras</i>	<i>Erro</i>	<i>Regras</i>	<i>Erro</i>	<i>Regras</i>
Binária	6.7%	3	6.7%	3	6.7%	3
Frequência	36.7%	2	36.7%	2	36.7%	2

¹⁴ Scripts disponíveis em <http://www.inf.unisinos.br/~cassiats/mestrado.htm>

Conforme pode ser observado nas Tabelas 3.1 e 3.2, a variação do número de palavras relevantes não implicou no aumento ou diminuição do erro de classificação, tanto nas ADs quanto nas RPs. Deste modo, pode-se verificar que os atributos que, de fato, caracterizam cada categoria estão entre os 20 mais frequentes de cada uma. Além disso, as árvores apresentaram menores taxas de erros, independentemente do tipo de codificação e número de palavras relevantes adotadas.

Os experimentos seguintes foram conduzidos com as RNAs, utilizando-se a ferramenta Neusim. Nestes experimentos, assim como nos iniciais, foram variados o número de palavras relevantes selecionadas (20, 50 e 80) e o tipo de codificação (binária e frequência). A RNA utilizada foi a MLP, sendo aplicados os algoritmos de aprendizado BP e CasCor.

As topologias das redes são formadas por 3 ou mais camadas. O número de neurônios na camada de entrada corresponde ao número de palavras no vetor global (dependente do número de palavras relevantes selecionadas); o número de neurônios da camada intermediária varia conforme o algoritmo de aprendizado utilizado; e o número de neurônios na camada de saída (3) corresponde a codificação binária atribuída para cada categoria (001, 010 e 100, para RNA, AG e SMA, respectivamente). A condição de parada do aprendizado foi o número máximo de 1000 épocas.

Nos experimentos com o algoritmo BP, foi utilizado um valor de 0.9 para o *momentum*, 0.1 para a taxa de aprendizado e variado o número de neurônios na camada intermediária (2, 4 e 8). Para cada topologia, foram realizadas 5 simulações, variando-se a semente aleatória e mantendo-se os demais parâmetros de configuração da rede.

A Tabela 3.3 apresenta os valores médios (para as 5 simulações) dos menores erros de generalização (melhores generalizações – *BestScore*) obtidas no processo de aprendizagem. Além disso, são apresentados os valores médios das melhores épocas de aprendizagem (*BestEpoch*), obtidas conforme alcance das melhores taxas de generalização, em cada simulação.

Conforme Tabela 3.3, a menor taxa de erro (1.3%) foi obtida utilizando-se 2 neurônios na camada intermediária, a codificação binária e um número entre 20 e 50 palavras relevantes. Quanto a pior taxa (12.0%), esta foi obtida com o uso da codificação por frequência, 80 palavras relevantes e 4 neurônios na camada

intermediária. Pode-se afirmar que 2 neurônios na camada intermediária são suficientes para o problema da categorização apresentado, oferecendo taxas de erro, em geral, inferiores as obtidas com o uso de 4 e 8 neurônios.

Tabela 3.3 – Média do menor erro de generalização (BP).

No. Neurônios	Codificação	No. Palavras por Categoria					
		20		50		80	
		<i>Erro</i>	<i>Épocas</i>	<i>Erro</i>	<i>Épocas</i>	<i>Erro</i>	<i>Épocas</i>
2	Binária	1.3%	91	1.3%	18	7.6%	17
	Frequência	8.6%	133	7.3%	24	8.6%	51
4	Binária	6.0%	6	2.7%	7	6.0%	78
	Frequência	6.7%	10	10.7%	5	12.0%	96
8	Binária	6.0%	13	2.7%	77	8.0%	151
	Frequência	7.3%	35	7.3%	123	7.3%	259

Os experimentos com o algoritmo CasCor foram realizados com um valor de *momentum* 0.0 e para a taxa de aprendizado 0.0001. Para cada variação nos parâmetros do experimento (codificação e número de palavras relevantes) foram realizadas 5 simulações, variando-se a semente aleatória e mantendo-se os demais parâmetros de configuração da rede. A Tabela 3.4 apresenta os valores médios (para as 5 simulações) dos menores erros de generalização (*BestScore*) e das melhores épocas de aprendizagem (*BestEpoch*).

Tabela 3.4 – Média do menor erro de generalização (CasCor).

Codificação	No. Palavras por Categoria					
	20		50		80	
	<i>Erro</i>	<i>Épocas</i>	<i>Erro</i>	<i>Épocas</i>	<i>Erro</i>	<i>Épocas</i>
Binária	3.4%	30	10.0%	20	6.0%	22
Frequência	6.0%	127	6.7%	12	6.7%	12

Observa-se, conforme Tabela 3.4, que o uso da codificação binária e 20 palavras relevantes apresentou a melhor taxa de classificação (3.4%). Entretanto, a utilização da codificação por frequência ofereceu valores estáveis (em torno de 6.0%). Em geral, não foram adicionados neurônios na camada intermediária. A exceção ocorreu com a utilização da codificação por frequência e 20 palavras relevantes por categoria onde foram adicionados, em média, 3 novos neurônios.

Com base nos resultados obtidos, foi feita uma comparação entre as técnicas. A Tabela 3.5 apresenta os parâmetros utilizados na obtenção das menores taxas de erros. Os valores considerados para o algoritmo BP foram os obtidos com o uso de 2 neurônios na camada intermediária.

Tabela 3.5 – Média do menor erro de generalização.

Técnica	Taxa	Codificação	No. Palavras
<i>Árvores de Decisão</i>	3.3%	Qualquer	qualquer
<i>Regras de Produção</i>	6.7%	Binária	qualquer
<i>RNA Backpropagation</i>	1.3%	Binária	20 e 50
<i>RNA Cascade Correlation</i>	3.4%	Binária	20

Conforme a Tabela 3.5, as melhores taxas de classificação foram obtidas com as RNAs com o algoritmo MLP-BP e as ADs. Embora o algoritmo BP tenha oferecido a menor taxa de erro (1.3%), as ADs apresentaram taxas próximas e mostraram-se robustas. Nos experimentos com as RNAs, para a obtenção das taxas de erro apresentadas, foram feitos ajustes finos nos parâmetros de treinamento, com base em um conjunto de experimentos preliminares. Com as ADs, a alteração dos seus parâmetros não implicou em variações bruscas das taxas de erro, sendo, então, utilizados os parâmetros *default* definidos na ferramenta correspondente. Deste modo, observou-se uma maior dependência das RNAs aos seus parâmetros do que a apresentada pelas ADs, e a necessidade de um número maior de experimentos com as RNAs, o que pode dificultar o processo de automatização da etapa de geração (aprendizado) do classificador. Além disso, a representação do conhecimento pelas ADs é de fácil compreensão, utilização e expansão, pois a árvore gerada pode ser facilmente convertida em regras do tipo IF-THEN e a adição de novas categorias implica na criação de novas regras. Ainda, quanto ao tipo de codificação adotado, apesar da codificação binária, em geral, implicar nas melhores taxas de erro, a codificação por frequência, além de agregar mais conhecimento (número de vezes que uma palavra ocorre em um documento e não apenas se a mesma ocorre ou não), apresentou menor variação nas taxas de erro, em relação ao número de palavras selecionadas, especialmente nas ADs. Por fim, quanto ao número de palavras relevantes, verificou-se que as palavras mais representativas de cada categoria estão entre as 20 mais frequentes selecionadas.

Portanto, conforme os resultados, as ADs apresentaram taxas de erro satisfatórias e mostraram-se adequadas ao processo de categorização proposto, sendo selecionadas para uso no processo. Quanto ao tipo de codificação e número de palavras relevantes, foram adotadas a codificação por frequência e 20 palavras para caracterizar cada categoria. Cabe destacar, entretanto, que melhorias nas taxas de erro podem ser obtidas através da aplicação de outros algoritmos de pré-processamento, e que estudos de algoritmos alternativos aos apresentados neste capítulo estão sendo estudados pelo Grupo de Inteligência Artificial¹⁵ (GIA) do Programa Interdisciplinar de Pós-Graduação em Computação Aplicada (PIPICA).

¹⁵ <http://www.inf.unisinos.br/gia-pipca.html>

4 Agentes Inteligentes

A abordagem de agentes vem sendo largamente discutida e adotada por diversas áreas da Ciência da Computação, tais como Inteligência Artificial (IA), Redes de Computadores (RC), Sistemas de Informação (SI) e Computação Gráfica (CG). Apesar de ser uma das abordagens centrais da IA, rapidamente proliferou-se para as demais áreas. Essa característica interdisciplinar reflete o estado atual da abordagem, onde não se tem um consenso sobre a definição de um agente. Cada grupo de pesquisa segue uma determinada linha, apresentando a sua definição de acordo com os seus objetivos – Goodwin (1994); Maes (1994); Hayes-Roth (1995); Russell and Norvig (1995); Wooldridge and Jennings (1995); Nwana (1996); Franklin and Graesser (1996); Brenner et al (1998); Tecuci (1998).

Este capítulo apresenta uma introdução à abordagem de agentes e está organizado como segue. A seção 4.1 sintetiza as várias definições usadas para o termo agente. A seção 4.2 apresenta as propriedades que caracterizam um agente inteligente. Na seção 4.3 são listadas as classificações atribuídas aos agentes. A seção 4.4 comenta sobre as arquiteturas de agentes. Por fim, na seção 4.5 são comentadas as aplicações de agentes inteligentes.

4.1 Definições

Em uma abordagem clássica, Wooldridge and Jennings (1995) distinguem dois usos gerais do termo agente. No primeiro, o termo é utilizado para denotar sistemas de hardware ou de software que empregam certo grau de autonomia, habilidade social, reatividade e pró-atividade. No segundo, particularmente adotado por pesquisadores da IA, o termo agente é empregado para denotar sistemas que, além de possuírem as propriedades acima citadas, são contextualizados ou implementados utilizando conceitos que são mais usualmente aplicados aos humanos, tais como conhecimento, crença, intenção e comprometimento.

Russell and Norvig (1995) definem um agente como um sistema capaz de perceber as informações do ambiente onde está inserido através de sensores e reagir através de atuadores. A Figura 4.1 apresenta uma representação genérica de um agente.

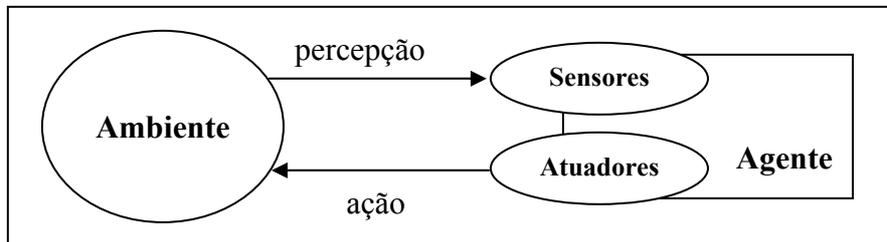


Figura 4.1 – Representação genérica de um agente [Russel and Norvig, 1995].

Segundo Brenner et al (1998), um agente pode ser definido como sendo um programa de software que pode realizar tarefas específicas e que possui um grau de inteligência que permite a ele realizar parte de suas tarefas autonomamente e interagir com o ambiente. Sob a visão de Tecuci (1998), um agente é um sistema baseado em conhecimento que percebe seu ambiente (uma interface gráfica de usuário, uma coleção de outros agentes, por exemplo); raciocina para interpretar percepções; resolve problemas e determina ações; e atua no ambiente para realizar um conjunto de tarefas para o qual foi designado. Conforme Garcia and Sichman (2003), agentes são personagens computacionais que atuam de acordo com um *script* definido, direta ou indiretamente, por um usuário.

Um agente pode interagir com um humano ou outro agente através de uma linguagem de comunicação, aceitar requisições, que indicam o que o usuário deseja, e decidir como satisfazer cada requisição, com algum grau de independência ou autonomia, exibindo um comportamento baseado em objetivos, escolhendo dinamicamente quais ações executar e em que ordem. Ele pode colaborar com os usuários na execução de suas tarefas, monitorar eventos ou procedimentos para o usuário, treiná-lo, ensiná-lo, ou auxiliá-lo na colaboração com outros usuários.

4.2 Características

Os agentes inteligentes podem ser caracterizados a partir de um conjunto de propriedades, as quais os diferenciam dos sistemas de software tradicionais. Wooldridge and Jennings (1995) descrevem algumas dessas propriedades, consideradas fundamentais:

- *autonomia*: agentes operam sem a intervenção direta de humanos ou de outros agentes e têm algum controle sobre suas ações e estado interno. Segundo Brenner et al (1998), a autonomia é a principal diferença entre um agente inteligente e um sistema de software tradicional. A possibilidade de agir de forma autônoma e de requerer a intervenção do usuário apenas em decisões importantes atribui aos agentes certo grau de inteligência.
- *habilidade social*: agentes interagem com outros agentes (humanos ou computacionais) através de uma linguagem de comunicação. A interação se dá, por exemplo, por necessidade de resolução de problemas, conveniência ou auxílio a outros agentes. Além disso, os agentes requerem uma interação com o ambiente onde estão inseridos, de forma a realizar eficientemente as suas tarefas.
- *reatividade*: agentes percebem o ambiente e reagem às alterações nele ocorridas.
- *pró-atividade*: agentes podem, além de agir em resposta as alterações no ambiente, exibir um comportamento dirigido a objetivos, tomando iniciativas quando julgarem apropriadas.

Franklin and Graesser (1996) citam outras propriedades:

- *adaptabilidade*: habilidade que os agentes possuem em adquirir conhecimento a partir de suas experiências e adaptar seu comportamento conforme o conhecimento adquirido.
- *temporalmente contínuo*: o agente é um processo que está continuamente em execução.
- *mobilidade*: capacidade dos agentes navegarem entre máquinas, em redes de computadores. Os agentes que não apresentam essa propriedade são caracterizados como estacionários.
- *flexibilidade*: habilidade dos agentes em agir quando as suas ações não estão descritas em roteiros pré-definidos.

- *personalidade*: esta propriedade refere-se aos agentes que possuem uma personalidade e um estado emocional.

Frozza (1997) considera, ainda, que os agentes são caracterizados por serem entidades reais ou virtuais; estarem em um ambiente; e possuírem metas, conhecimento e raciocínio. Por fim, os agentes podem possuir uma representação gráfica 2D ou 3D.

Deste modo, os agentes inteligentes podem ser considerados softwares com determinadas propriedades, sendo a interação com o ambiente uma propriedade fundamental.

4.3 Classificação

Assim como não existe um consenso sobre a definição do termo agente, não se tem um formato padrão para classificá-los. Inúmeras formas de classificação são encontradas na literatura.

Uma forma padrão de classificação, geralmente adotada, está baseada no subconjunto de propriedades que os agentes possuem. Pode-se atribuir uma classificação binária, considerando apenas uma propriedade, ou uma classificação múltipla, adotando várias delas. Por exemplo, seguindo uma classificação binária, pode-se dizer que um agente é móvel ou estacionário ou, seguindo uma classificação múltipla, indicar que o agente é móvel e não-adaptativo, dadas as propriedades de mobilidade e adaptabilidade. Entretanto, cada agente satisfaz as propriedades de autonomia, habilidade social, reatividade e pró-atividade [Jennings and Wooldridge, 1996] e de ser temporalmente contínuo [Franklin and Graesser, 1996]. Adicionando outras propriedades às fundamentais, são produzidas as classificações.

Além disso, existem outras possibilidades de classificação, de acordo com: (a) a tarefa que o agente executa (para filtro de e-mail ou de informações, por exemplo); (b) arquitetura de controle (reativos, cognitivos, híbridos, baseados em estados mentais); (c) ambiente onde estão inseridos (web, base de dados, outros); e (d) tipo de conhecimento que possuem (preferências e interesses do usuário, informações sobre o negócio).

Alguns autores propõem classes ou denominações aos agentes. Franklin and Graesser (1996) apresentam uma classificação ampla, baseada em uma taxionomia biológica, dividindo os agentes em biológicos, robôs e computacionais. Os agentes

computacionais são divididos em agentes de vida artificial e de software, sendo os últimos classificados em agentes baseados em tarefas, de entretenimento e vírus.

Jennings and Wooldridge (1996) distinguem três classes, de acordo com o nível de sofisticação dos agentes: “*gopher*” (executam tarefas baseadas em regras pré-especificadas); “*service performing*” (realizam uma tarefa a partir de uma requisição do usuário); e “*predictive*” (disponibilizam informações ou executam ações para o usuário, com certo grau de voluntariedade). Além disso, denominam agentes de usuários, os que possuem conhecimento sobre as preferências e interesses dos usuários, e agentes de negócios, os que detêm informações sobre o negócio (serviços, produtos, entre outras).

Brenner et al (1998) distinguem três categorias, dependendo da tarefa que os agentes executam: informativos (que oferecem suporte ao usuário na busca de informações em fontes distribuídas); cooperativos (que atuam na resolução de problemas complexos, através de cooperação e comunicação com outros objetos, agentes ou fontes externas); e transacionais (cujas tarefas principais são processar e monitorar processos).

Reilly and Bates (1992) apresentam os agentes emocionais (possuem um modelo de emoções e personalidade); Maes (1994) cita os agentes autônomos (realizam suas tarefas de forma essencialmente autônoma); Sycara et al (1996) apresentam os agentes de interface (que interagem diretamente com o usuário) e os de tarefa (auxiliam usuário na realização de tarefas, formulando planos para a resolução de problemas).

Nwana (1996) cita os colaborativos (ênfases na cooperação com outros agentes); móveis, reativos (agem em resposta a estímulos do ambiente, sem capacidade de raciocínio), e híbridos (possuem características dos agentes reativos e capacidade de raciocínio); Rickel and Johnson (1997) citam os agentes pedagógicos (atuam em ambientes educacionais); e Aylett and Cavazza (2000) definem os agentes virtuais (atuam em ambientes virtuais e possuem representações gráficas associadas), que podem ser classificados como agentes físicos (ênfase dada ao comportamento físico do agente no ambiente) ou cognitivos (ênfase atribuída às habilidades cognitivas do agente e à interação com o usuário do sistema).

Por fim, Garcia and Sichman (2003), apresentam uma taxionomia na qual um agente é considerado um tipo especial de sistema computacional e que pode ser

classificado segundo alguns eixos: cognitivo, de foco, de atuação e ambiental. No eixo cognitivo, os agentes podem agir baseados em modelos racionais de decisão (cognitivo) ou apenas em modelos de reação aos estímulos provocados pelo ambiente (reativo). No eixo de foco, eles enfatizam similaridades físicas (estrutural) ou comportamentais (comportamental) com humanos. Quanto ao eixo de atuação, os agentes podem atuar de forma isolada (isolado) ou com outros agentes (social). Por fim, o eixo ambiental divide os agentes em atuantes em *desktop* (agente de *desktop*) ou em uma rede internet ou Intranet (agente de internet).

A Tabela 4.1 apresenta uma síntese das principais classificações atribuídas aos agentes inteligentes. São apresentados os critérios que podem ser utilizados para classifica-los, juntamente com os tipos correspondentes.

Tabela 4.1 – Classificações de agentes inteligentes.

Critério	Classificação
Tipo de entidade	Real (humano, biológico, robô físico) ou computacional (de vida artificial, de software)
Tipo de similaridade com humanos	Estrutural (físico) ou comportamental
Arquitetura de controle	Reativo, cognitivo, híbrido, baseado em estados mentais, com modelo de emoções
Tarefa	Transacional, informativo, de negócio, de usuário, de interface
Grau de autonomia	Autônomo, semi-autônomo, não autônomo (controlado completamente pelo usuário)
Localização	Móvel, estacionário, distribuído
Ambiente de atuação	De <i>desktop</i> (ambiente fechado), de rede (ambiente aberto), pedagógico (ambiente educacional), virtual (ambiente virtual tridimensional)
Tipo de atuação	Isolada ou social (grupo, cooperativo ou não cooperativo)
Tipo de interação	Com usuário, com outros agentes, com o ambiente, múltiplo

Além destas possibilidades de classificação e denominações de agentes, inúmeras outras existem na literatura, as quais, muitas vezes, combinam as possibilidades apresentadas nessa seção.

4.4 Arquiteturas

Uma arquitetura de agentes indica uma metodologia para a implementação de agentes. Através dela, é especificado como um agente se comporta durante a interação com o ambiente e na realização de suas tarefas. Segundo Correa (1994), para definir a arquitetura de um agente é necessário conhecer o tipo de tarefa que o agente irá realizar e o seu papel no ambiente onde se encontra.

Essas arquiteturas podem ser classificadas de acordo com os mecanismos usados pelos agentes para selecionar uma ação [Viccari and Giraffa, 1996]. Em função disso, pode-se classificar as arquiteturas de agentes em: cognitivas, reativas, híbridas e baseadas em estados mentais. Nas seções que seguem, tais arquiteturas são comentadas.

4.4.1 Arquitetura Reativa

Uma arquitetura é denominada reativa ou não-deliberativa quando a escolha da ação a ser executada está relacionada de forma direta com a ocorrência de eventos no ambiente. Nessa arquitetura, o controle das ações do agente é realizado a partir de um comportamento do tipo situação – ação (ou estímulo – resposta). O agente age em um espaço de tempo, com base em uma pequena quantidade de informação, no instante em que recebe ou percebe algum sinal ou estímulo do ambiente.

Nesse tipo de arquitetura não há representação explícita do conhecimento sobre o ambiente [Brenner et al, 1998]. O conhecimento dos agentes é implícito e manifestado através de comportamentos, o que pode restringir a autonomia do agente e sua capacidade de aprender e melhorar seu desempenho. Outra característica marcante dessa arquitetura é a ausência de memória das ações passadas, sendo que o resultado de uma ação passada não exerce influência direta sobre as ações futuras.

Os agentes modelados a partir de uma arquitetura reativa, denominados reativos ou não-deliberativos, não possuem capacidade de raciocínio e planejamento, e por isso, são consideradas entidades mais simples do que os agentes cognitivos. São agentes baseados em comportamento, definido a partir da situação atual do ambiente e do conhecimento (entrada sensorial) que possuem. Além disso, as ações destes agentes são realizadas em resposta a estímulos oriundos do ambiente.

Como exemplos destes agentes, podem ser citados os robôs com uma arquitetura de controle reativa. Eles possuem um número de sensores que os permitem perceber o ambiente, e as informações recebidas pelos sensores são utilizadas para disparar a próxima ação no ambiente. Por exemplo, sempre que um sensor detecta um obstáculo, uma alteração na direção do movimento do robô deve ser realizada. É o típico caso “situação – ação”.

4.4.2 Arquitetura Cognitiva

Uma arquitetura é dita cognitiva ou deliberativa quando a escolha da ação a ser executada pelo agente é realizada a partir de um modelo simbólico do ambiente e de um plano de ações. Essa arquitetura está fundamentada na produção de uma seqüência de ações (planos) para alcançar um determinado objetivo. Tais ações estão baseadas nas hipóteses de que o agente possui um conhecimento do ambiente e de outros agentes. Para isso, é mantida uma representação explícita do conhecimento sobre o ambiente, bem como um histórico das ações passadas.

Entretanto, a arquitetura cognitiva é tipicamente incapaz de agir rápida e adequadamente perante situações não previstas [Giraffa, 1997]. Ela adota a hipótese de que as condições do mundo permanecem estáticas enquanto o agente estiver executando as suas ações ou processando alguma informação para deliberar sobre as ações. Por outro lado, essas arquiteturas apresentam objetivos explícitos, que podem ser alterados, bem como componentes de percepção, aprendizagem e raciocínio.

Os agentes modelados a partir de uma arquitetura cognitiva, denominados cognitivos ou deliberativos, raciocinam e decidem sobre quais objetivos devem alcançar, que planos seguir e quais ações devem ser executadas em um determinado momento. Deste modo, um agente executa uma ação inteligente quando, possuindo um certo objetivo e o conhecimento de que uma certa ação o conduzirá a este objetivo, seleciona esta ação.

Os agentes cognitivos agem de acordo com o seu conhecimento, porque dispõem de uma capacidade de raciocínio sobre uma base de conhecimento e aptidões para tratar de informações diversas. Tais informações estão ligadas ao domínio da aplicação e são relativas às interações entre os agentes e o ambiente [Frozza, 1997]. Além disso,

planejam ações futuras a partir do raciocínio aplicado à memória das ações realizadas no passado.

4.4.3 Arquitetura Híbrida

A arquitetura híbrida é aquela em que a escolha da ação é realizada usando uma combinação entre as técnicas utilizadas em arquiteturas cognitiva e reativa [Giraffa, 1997]. Essa arquitetura foi proposta como alternativa para solucionar as deficiências principais das duas arquiteturas anteriores. A arquitetura cognitiva é tipicamente incapaz de agir rápida e adequadamente perante situações não previstas. Na arquitetura reativa, o agente é incapaz de descobrir alternativas para o seu comportamento quando a situação do mundo diverge bastante de seus objetivos iniciais. Além disso, o agente não possui capacidade de raciocínio e planejamento.

O objetivo é construir um agente atuante em dois subsistemas: o sistema cognitivo, que contém um modelo simbólico do mundo, utilizando planejamento e tomada de decisões, e o sistema reativo, capaz de reagir a eventos que ocorrem no ambiente.

Os agentes híbridos são normalmente projetados através de uma arquitetura hierárquica. Os níveis mais baixos representam o sistema reativo e são usados para a aquisição de informações do ambiente, de outros agentes ou de outras fontes. Os componentes cognitivos, responsáveis pelo planejamento e determinação de objetivos, são usados nos níveis mais altos [Brenner et al, 1998].

4.4.4 Arquitetura Baseada em Estados Mentais

A idéia básica de uma arquitetura baseada em estados mentais está em descrever o processamento interno de um agente utilizando um conjunto básico destes estados, tais como crenças, desejos, intenções, expectativa, entre outros. A arquitetura BDI (*Belief, Desire, Intention*) é um exemplo de arquitetura baseada em estados mentais, apesar de ser considerada por alguns autores uma estrutura deliberativa, pelo fato de manter uma representação simbólica do ambiente, expressa em crenças, desejos e intenções.

Segundo Rao e Georgeff (1995), *Belief* (crença), *Desire* (desejo) and *Intention* (intenção) representam, respectivamente, a informação, a motivação e o estado deliberativo de um agente. De modo intuitivo, as crenças correspondem à informação

que o agente possui sobre seu ambiente, desejos representam opções de estados futuros disponíveis ao agente (motivação) e intenções são os estados futuros que o agente escolheu e com os quais comprometeu-se (deliberação).

As crenças, segundo Brenner et al (1998) são as visões do agente diante do mundo, as expectativas que o agente possui para os estados futuros. É o conhecimento do ambiente, representado de forma explícita. Para Rao and Georgeff (1995) é o componente informativo do estado do sistema, necessário para fornecer informações sobre o provável estado do ambiente.

Os desejos representam o julgamento dos estados futuros [Brenner et al, 1998]. O agente pode querer que um estado contido em suas crenças possa ocorrer no futuro. Eles representam o estado motivacional do sistema [Rao and Georgeff, 1995]. Um desejo é uma noção abstrata que indica as preferências sobre os estados futuros do ambiente. Representa uma situação ou conjunto de situações em que o agente gostaria que o mundo estivesse.

As intenções são as metas escolhidas conforme a prioridade do agente. Elas representam os componentes deliberativos do sistema [Rao and Georgeff, 1995], que servem para decidir o curso de ação que deve ser tomado pelo sistema. É o resultado das escolhas, que leva o agente a uma ação.

Neste contexto, o raciocínio prático do agente envolve repetidamente atualizar as crenças a partir da percepção do ambiente, decidir quais opções estão disponíveis, filtrar estas opções para determinar novas intenções e agir baseado nestas intenções. A idéia é decidir qual ação executar que melhor atenda os objetivos dos agentes. Em suma, envolve decidir quais metas devem ser atingidas e como isso será feito.

4.4.5 Outras Arquiteturas

No contexto de sistemas multiagentes, a metodologia AEIO [Demazeau, 1995] considera o problema a ser modelado como composto por quatro elementos: *agents*, *environment*, *interactions* e *organization*. Na definição desses elementos, são considerados, respectivamente: o número de agentes e seus tipos (cognitivos ou reativos, por exemplo); as características do ambiente (dependentes do domínio do

problema); as interações entre os agentes; e os mecanismos de coordenação e cooperação entre os agentes.

Outra metodologia que considera a interação entre agentes é a KSI [Musse, 2000], utilizada na modelagem de agentes virtuais em ambientes que envolvem multidões. Tal metodologia é composta por três tipos de informações: *knowledge* (conhecimento que o agente possui sobre os outros agentes, o ambiente e as ações passadas), *status* (atributos individuais, tal como estado emocional da entidade) e *intentions* (objetivos do agente, tais como ir a algum lugar, comunicar-se, outros).

4.5 Aplicações

A abordagem de agentes está sendo utilizada em uma variedade de aplicações, compreendendo desde sistemas relativamente simples, tais como filtros de informações, até sistemas críticos, como controles de tráfego aéreo. Alguns dos domínios e exemplos cuja abordagem vêm sendo aplicada estão listados abaixo [Brenner et al, 1998; Musse, 2000; Santos et al, 2002; Garcia and Sichman, 2003]:

- *industrial*: controle de processos; manufatura (gerencia do processo de produção); controle de tráfego aéreo; gerenciamento de redes (monitoramento da rede e controle de congestionamentos); e telecomunicações;
- *comercial*: gerenciamento de informações (recuperação e filtro de informações); comércio eletrônico (agentes que procuram especificações, preços, e serviços requeridos pelo usuário ou que atuam em processos de negociação com agentes de negócio, por exemplo); gerenciamento de bases de dados (manutenção e atualização da base);
- *entretenimento*: jogos eletrônicos (personagens do jogo modelados como agentes); espaços virtuais de interação (shoppings e teatros virtuais, onde agentes povoam ambientes, por exemplo); personagens artificiais (que provêm interação com os usuários);
- *educação*: treinamento e ensino a distância (acompanhamento da interação do aluno com o sistema educacional; auxílio ao aluno durante a interação com o sistema; coleta de dados sobre o modelo do aluno, utilizado

para personalização da interface do sistema; seleção de estratégias de ensino, conforme modelo; entre outras atividades).

- *simulação*: investigação de agentes virtuais e seus comportamentos em espaços urbanos; simulação comportamental de multidões de agentes virtuais em situações de emergência.

Conforme visto neste capítulo, a abordagem de agentes vêm sendo adotada nas mais diversas áreas e aplicações. Atualmente, atenção tem sido atribuída ao uso de agentes inteligentes em ambientes virtuais. Nestes ambientes, o agente pode atuar como assistente do usuário na exploração do ambiente, sendo dada ênfase para certas propriedades dos agentes inteligentes, tais como percepção, habilidade social, reatividade, capacidade de raciocínio, adaptabilidade e representação gráfica. A partir da interação com o usuário (habilidade social), de sua percepção, capacidade de raciocínio e reatividade, o agente reconhece as solicitações do usuário, decide como agir e atua no ambiente (através da movimentação de sua representação gráfica). Com base em sua adaptabilidade, é capaz de atualizar o seu conhecimento a partir das modificações que ocorrem no ambiente.

No próximo capítulo, são tratados os Ambientes Virtuais Inteligentes (AVIs), os quais utilizam os agentes inteligentes como um dos seus componentes principais.

5 Ambientes Virtuais Inteligentes

Realidade Virtual (RV) é uma técnica avançada de interface através da qual o usuário pode imergir, navegar e interagir em um ambiente sintético tridimensional, utilizando canais multisensoriais. Segundo Teichrieb (1999), a RV tornou-se uma alternativa atraente para o desenvolvimento de interfaces virtuais mais realistas e interessantes para o usuário. Os ambientes que fazem uso de técnicas de RV são comumente referenciados como Ambientes Virtuais (AVs).

Atualmente, as pesquisas em Inteligência Artificial (IA), Vida Artificial (VA) e AVs têm sugerido uma integração entre estas áreas. O objetivo é a criação de ambientes virtuais que exploram o uso de entidades com certo grau de inteligência e os efetivos meios de suas representações gráficas, provendo maior dinamicidade, realismo e usabilidade dos ambientes. Conforme Aylett and Luck (2000), os ambientes que exploram tal integração são denominados Ambientes Virtuais Inteligentes (AVIs) e têm derivado uma nova área de pesquisa.

Para Aylett and Cavazza (2001), a inclusão de uma camada de IA em um ambiente virtual pode ser justificada sob várias perspectivas: adição de um componente para resolução de problemas para o ambiente virtual, como por exemplo, o planejamento de ações em aplicações interativas; agregação de um nível de conhecimento para suportar a representação conceitual do ambiente; aumento da interatividade no ambiente, por reconhecer interações com o usuário e determinar comportamentos adaptativos do sistema; e permitir a descrição de comportamentos causais como uma alternativa para a simulação física. A última perspectiva é especialmente adotada por pesquisadores da área de VA. Neste caso, a integração com os AVs envolve a simulação de modelos de animação, utilizados para agregar comportamentos a entidades virtuais, geralmente animais ou humanos virtuais.

Desta forma, a área de AVIs abrange diferentes grupos de pesquisa concentrados em diversos aspectos do ambiente e da inteligência embutidos neste. Uma conceituação genérica para os AVIs é proposta por Anastassakis et al (2001). Segundo estes pesquisadores, um AVI pode ser definido como um ambiente virtual semelhante a um

mundo real, habitado por entidades autônomas inteligentes exibindo uma variedade de comportamentos. Estas entidades podem ser objetos estáticos simples ou dinâmicos, representações virtuais de formas de vida (humanos ou animais), avatares (que representam usuários no ambiente) e outros.

Conforme Rickel et al (2002), Gratch et al (2002) e Anastassakis et al (2001), as aplicações potenciais destes ambientes são consideráveis, podendo ser empregados em uma variedade de áreas, principalmente relacionadas com a simulação, o entretenimento e a educação. Em simulação, ambientes de diferentes tipos (espaços urbanos abertos ou interiores, habitados por humanos virtuais) podem auxiliar em projetos arquitetônicos e no controle de tráfego de pessoas ou carros. Na área de entretenimento, os jogos (com cenários que podem ser adaptados conforme o andamento do jogo), os teatros e as lojas virtuais (onde o usuário pode navegar e interagir com outros usuários) podem ser citados como aplicações potenciais. Por fim, em sistemas educacionais, a incorporação de personagens tutores e sofisticadas técnicas de representações da informação provêm uma efetiva e agradável experiência de aprendizado.

Uma das principais áreas da IA que tem recebido especial atenção em AVIs é a de agentes inteligentes, comentada no capítulo 4. Tais agentes, quando inseridos em AVIs são denominados Agentes Virtuais Inteligentes [Aylett and Cavazza, 2001] e podem assumir uma variedade de papéis: humanos [Rickel and Johnson, 1997]; criaturas fictícias [Grand and Cliff, 1998] ou animais [Reynolds, 1987; Terzopoulos et al, 1994]. Nos ambientes onde os agentes interagem com os usuários, eles atuam como assistentes na exploração do ambiente e localização de informações, podendo estabelecer uma comunicação verbal (em linguagem natural, por exemplo) ou não verbal (através de movimentação, gestos e expressões faciais) com o usuário. Podem, ainda, executar ações no ambiente, conforme solicitações do usuário. Eles possuem uma representação de um corpo, o qual deve mover-se pelo ambiente de uma maneira fisicamente convincente, e algum comportamento associado. Além disso, de forma que possam atuar no ambiente em que estão inseridos, devem manter um acoplamento entre os seus comportamentos e o estado do ambiente virtual.

Por fim, outra área da IA cujas técnicas estão sendo aplicadas aos AVIs é a de Processamento de Linguagem Natural (PLN). O objetivo é explorar uma forma conveniente de acessar informação e controlar o ambiente virtual. A comunicação em

linguagem natural para instruir a usuário na navegação no ambiente, provendo assistência e explicações, e para controlar agentes virtuais (para realização de determinada ação no ambiente) tem sido adotada.

Este capítulo objetiva apresentar alguns dos principais trabalhos propostos na área de Ambientes Virtuais Inteligentes. Na seção que segue, tais trabalhos são apresentados. É atribuída ênfase aos ambientes que agregam agentes virtuais que interagem com o usuário. Ao final da seção um comparativo entre os ambientes é apresentado.

5.1 Trabalhos Relacionados

O Active Worlds¹⁶ é talvez o mais conhecido conjunto de ambientes virtuais tridimensionais, disponível comercialmente. Nestes ambientes, o usuário, representado por um avatar, pode navegar e interagir com outros usuários. Agentes virtuais, que se deslocam no ambiente de forma pré-programada (trajetória fixa), são utilizados para popular os ambientes sem, no entanto, serem capazes de interagir com o usuário para a troca de informações. O usuário pode, apenas, controlar alguns movimentos (caminhar e dançar) dos avatares. As Figuras 5.1 (a) e (b) apresentam interfaces de ambientes do Active Worlds.



(a)



(b)

Figura 5.1 – Interfaces de ambientes do Active Worlds.

Um agente virtual inteligente, STEVE (*Soar Training Expert for Virtual Environment*), que atua como tutor em cursos de treinamento é proposto por Rickel and

¹⁶ <http://www.activeworlds.com>

Johnson (1997). O treinamento acontece em um espaço tridimensional interativo, onde STEVE auxilia o usuário no aprendizado de uso de um equipamento. As Figuras 5.2 (a) e (b) ilustram o agente STEVE no ambiente em que atua.

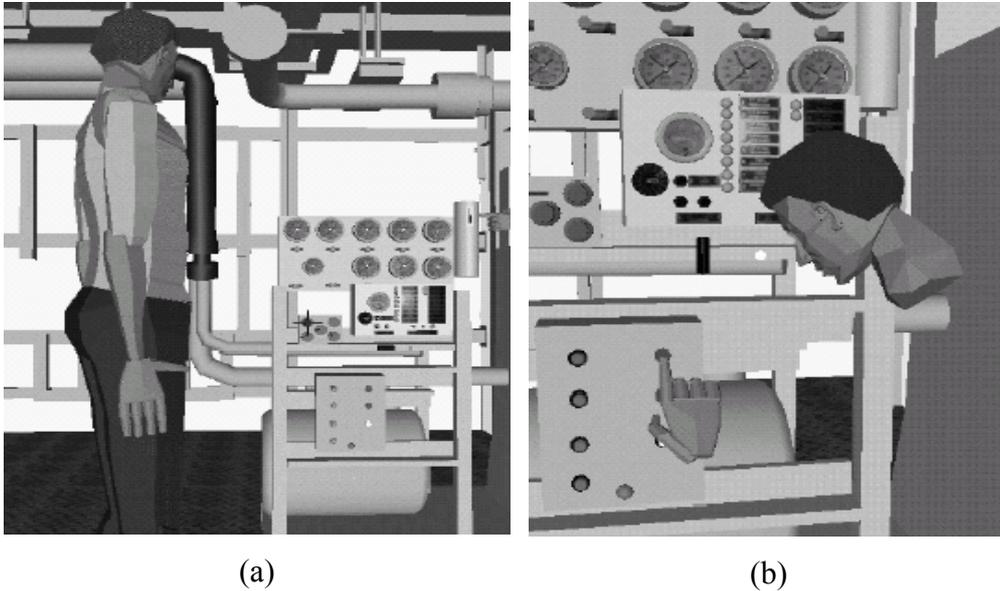


Figura 5.2 – (a) STEVE ao lado de um equipamento; (b) STEVE demonstrando como utilizar o equipamento [Rickel and Johnson, 1997].

Bersot et al (1998) apresentam Ulysses, um agente de conversação inserido em um ambiente virtual. O agente auxilia os usuários na navegação pelo ambiente e aceita comandos, em linguagem natural, para a manipulação de objetos. Além da movimentação da representação gráfica do agente pelo ambiente, é utilizado um sintetizador de fala para responder as solicitações do usuário. As Figuras 5.3 (a) e (b) ilustram a movimentação do agente pelo ambiente.

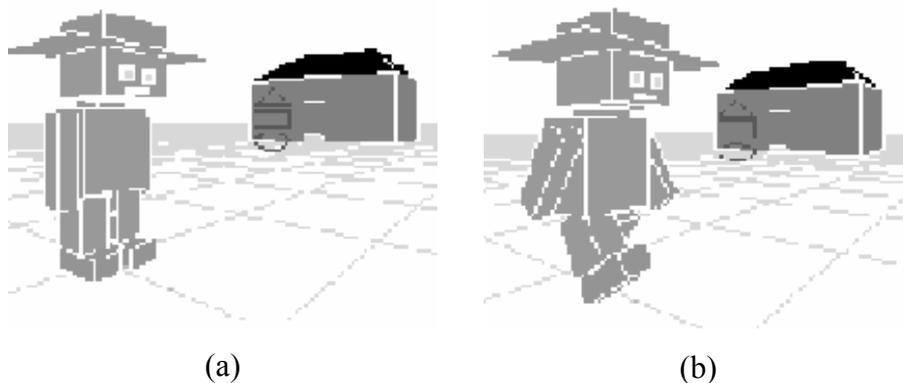


Figura 5.3 – Agente Ulysses caminhando pelo ambiente [Bersot et al, 1998].

Um guia virtual, George, que conduz visitantes a lugares relevantes em uma universidade virtual, de acordo com as necessidades de informação dos mesmos, bem como apresenta documentos multimídia apropriados, é proposto por Panayiotopoulos et al (1999). O agente virtual se comunica com os usuários, os quais podem indicar suas necessidades de informação através de linhas de comando. As Figuras 5.4 (a) e (b) apresentam a interface de entrada ao ambiente e a movimentação do agente.

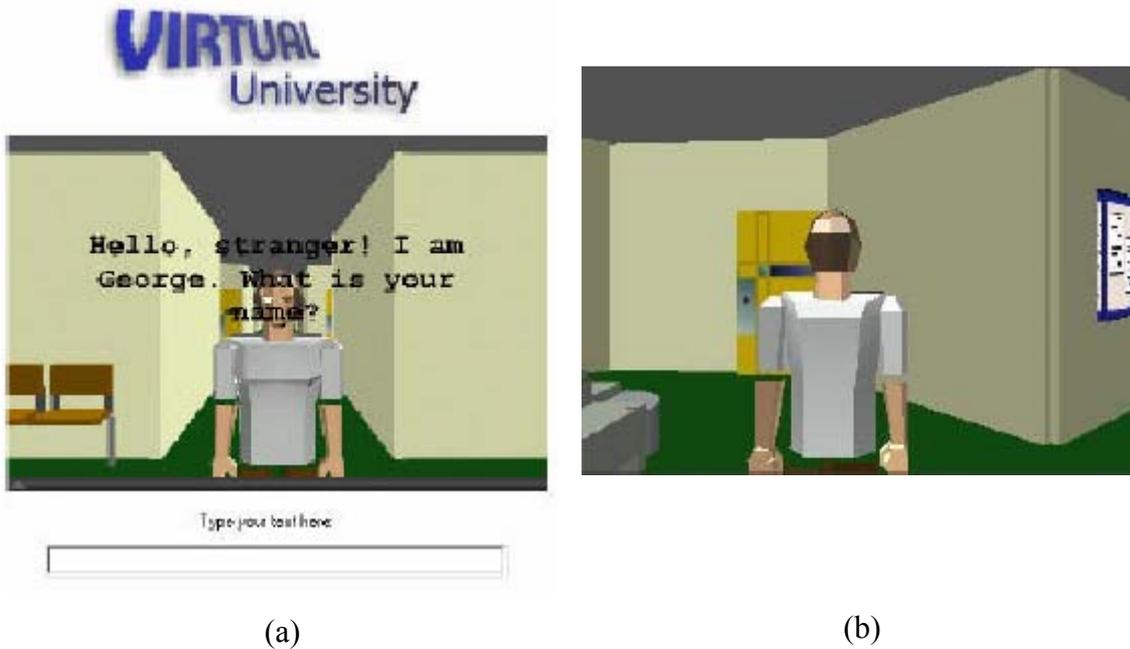


Figura 5.4 – (a) Interface de entrada ao ambiente; (b) Agente se deslocando no ambiente [Panayiotopoulos et al, 1999].

Noll et al (1999) apresentam o GuideAgent, um agente virtual cujas tarefas são navegar através de um ambiente tridimensional, prover informações sobre os objetos, os usuários e o próprio ambiente, e auxiliar o usuário na exploração do mesmo. O agente e os usuários são representados graficamente no ambiente e o agente é capaz de modelar o comportamento e as preferências do usuário.

Um teatro virtual é apresentado por Nijholt and Hulstijn (2000). No ambiente, construído em VRML (*Virtual Reality Modeling Language*), os usuários podem navegar, visitando salas de concerto e admirando quadros, e interagir com um agente virtual informacional. O agente possui informações sobre shows, músicos e ingressos, comunica-se com o usuário através de diálogo em linguagem natural, e apresenta diferentes expressões faciais. O agente pode, ainda, auxiliar o usuário na navegação

pelo ambiente. As Figuras 5.5 (a) e (b) apresentam a interface externa do teatro e o agente virtual, respectivamente.

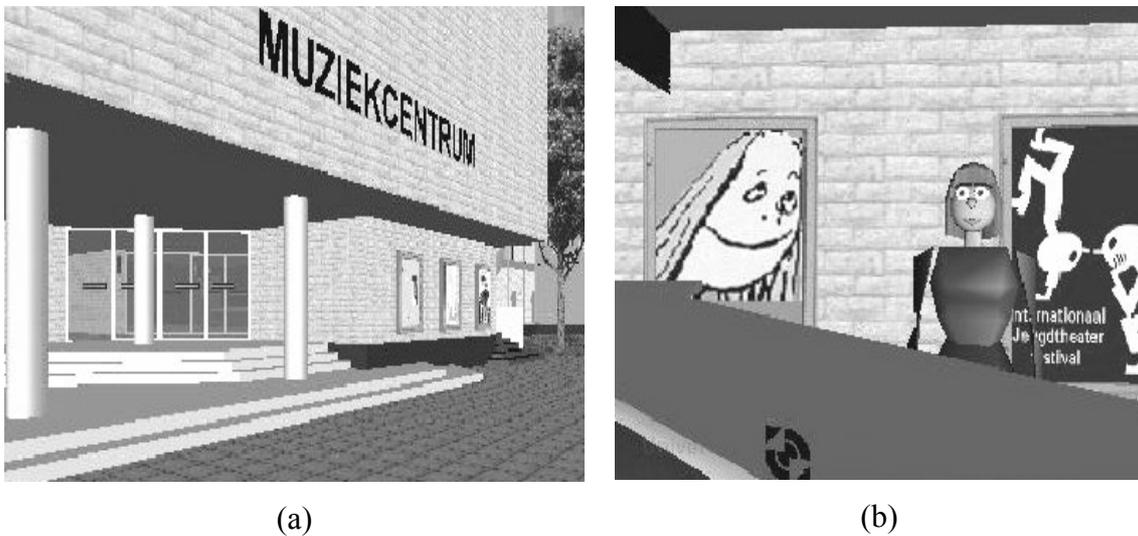


Figura 5.5 – (a) Entrada do teatro virtual; (b) Agente virtual [Nijholt and Hulstijn 2000].

O sistema “Lokutor”, proposto por Milde (2000), é um ambiente tridimensional onde um agente de apresentação comunica-se com o usuário através de diálogo em linguagem natural. A tarefa do agente é apresentar produtos aos usuários, atuando como um assistente de venda. A entrada em linguagem natural é analisada por um *parser*, responsável por extrair características relevantes do diálogo e repassá-las a um sistema deliberativo, encarregado pela definição do comportamento do agente e determinação da próxima ação. Os aspectos de diálogo do agente estão baseados na utilização de um manual do produto. As Figuras 5.6 (a), (b), (c) e (d) ilustram a seqüência de ações do agente durante a apresentação de um produto ao usuário.

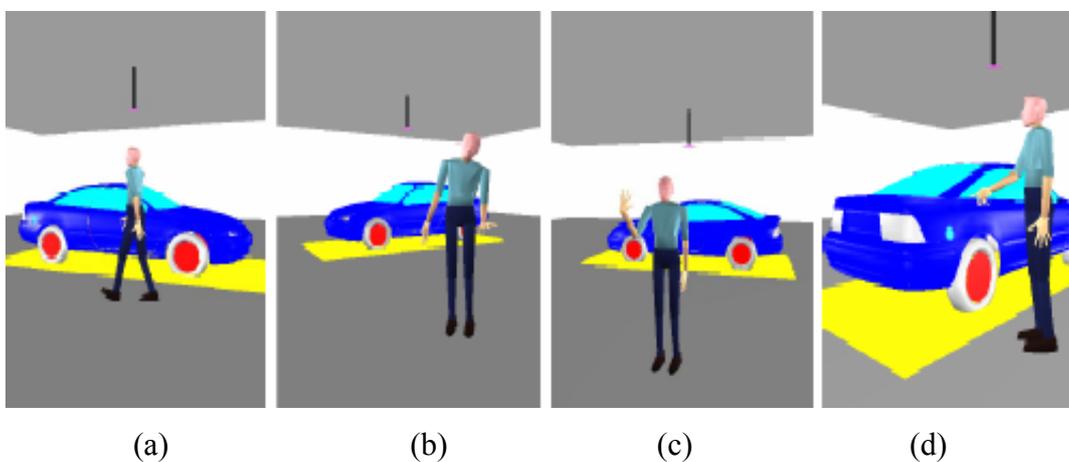


Figura 5.6 – Apresentação do produto pelo agente [Milde, 2000].

Anastassakis et al (2001) apresentam um ambiente habitado por agentes que o exploram, localizam itens específicos, interagem uns com os outros e respondem a instruções do usuário (como por exemplo, ir a um determinado local). No protótipo apresentado, o qual consiste em uma biblioteca virtual construída em VRML, dois agentes (cliente e bibliotecário) se comunicam. O agente cliente requisita informações sobre livros ao agente bibliotecário, responsável por localizar e apresentar o item desejado. A Figura 5.7 apresenta a interface do ambiente prototipado.



Figura 5.7 – Interação entre os agentes cliente e bibliotecário [Anastassakis et al, 2001].

Uma metodologia para navegação e exploração de ambientes virtuais tridimensionais é descrita por Frery et al (2002). São utilizados avatares como guias interativos, que possuem estratégias de navegação baseadas na informação que detém sobre o ambiente. A inteligência dos avatares é representada através dos seus comportamentos e do conhecimento que possuem do ambiente e do usuário. Com base no conhecimento sobre o usuário, coletado a partir de formulários, é feita a estruturação do conteúdo a ser apresentado e a modificação da representação gráfica dos avatares.

A aplicação de um AVI no tratamento de crianças hiper-ativas é apresentada por Rizzo et al (2002). O ambiente consiste de uma sala virtual tridimensional onde um avatar representa uma professora e objetos se movimentam pelo ambiente. O uso de um dispositivo para imersão no ambientes (HMD¹⁷ - *Head Mounted Display*) permite monitorar os movimentos da criança, detectando o seu campo de visão. Os dados

¹⁷ Dispositivo de saída (vídeo-capacetes) utilizado em aplicações de Realidade Virtual. É o dispositivo que permite uma maior imersão do usuário no mundo virtual. A presença de sensores permite o rastreamento da posição e orientação da cabeça do usuário.

coletados a partir da monitoração são utilizados para análise do comportamento da criança. A Figura 5.8 ilustra a sala de aula virtual.



Figura 5.8 – Sala de aula virtual [Rizzo et al, 2002].

Além da sala de aula virtual, Rizzo et al (2002) utilizam um escritório virtual com diversos objetos dinâmicos (telefones que tocam, computadores e vários avatares), para a verificação de aspectos cognitivos (memória, percepção e atenção) dos seus usuários. Os avatares, que se movimentam pelo ambiente, representam supervisores e chefes, e podem solicitar ao usuário a realização de determinada atividade no ambiente. O dispositivo HMD é utilizado para monitorar os movimentos do usuário na realização das atividades. Além disso, o ambiente é modificado e os aspectos de percepção do usuário quanto às modificações são coletados. A Figura 5.9 ilustra as várias configurações do escritório virtual.



Figura 5.9 – Várias configurações do escritório virtual [Rizzo et al, 2002].

Um ambiente virtual tridimensional adaptativo, desenvolvido em VRML, é apresentado por Chittaro and Ranon (2002, 2002a). No ambiente, que consiste em uma loja virtual, os usuários podem navegar e obter informações sobre os produtos nele espalhados. As informações sobre os interesses e características dos usuários, utilizadas para a personalização do ambiente, são coletadas através de formulários e monitoração

das ações do usuário no ambiente (tais como produtos visualizados e compras efetuadas). Objetos que se deslocam no ambiente auxiliam o usuário na navegação e localização de produtos específicos. As Figuras 5.10 (a) e (b) apresentam as adaptações no ambiente e os objetos que trafegam pelo mesmo.



Figura 5.10 – Adaptações na loja virtual [Chittaro and Ranon, 2002; 2002a].

Em um trabalho recente, Chittaro et al (2003), apresentam um agente virtual designado a auxiliar o usuário na navegação por um museu virtual. A partir da descrição dos lugares ou objetos de interesse a serem visitados no museu, o agente cria uma trajetória apropriada. Além disso, o agente está apto a parar durante o trajeto e apresentar cada objeto ou lugar. As Figuras 5.11 (a) e (b) apresentam o agente em uma explanação sobre um dos objetos do ambiente e uma trajetória criada, respectivamente.

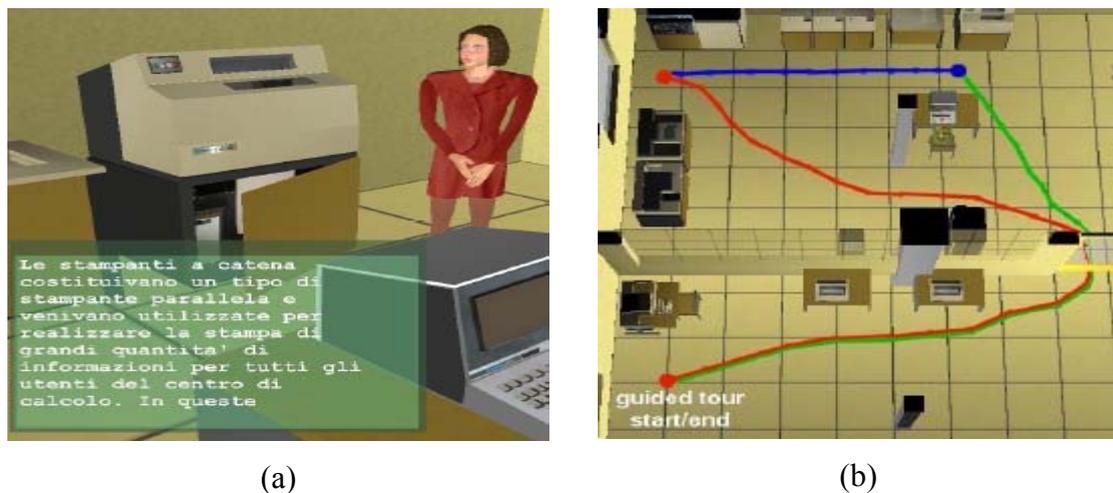


Figura 5.11 – (a) Agente explanando sobre o objeto; (b) Trajetória elaborada para a navegação pelo ambiente [Chittaro et al, 2003].

Nesta seção, foram apresentados os trabalhos que propõem a utilização de ambientes virtuais inteligentes, para variados propósitos. Como forma de sintetizar as funcionalidades providas por cada um, um comparativo entre os ambientes está demonstrado na Tabela 5.1. Cabe salientar que, no item *Ambiente*, é indicado o valor “dinâmico” para os casos em que o ambiente apresenta a possibilidade de movimentar objetos inseridos nele ou possui objetos que se movimentam; e “estático”, caso não apresente estas possibilidades. Além disso, para o item *Agente*, indica-se o valor “único” para os ambientes que possuem apenas um agente; e “vários” para os que apresentam múltiplos agentes interagindo.

Com base nos requisitos utilizados para caracterizar cada ambiente, pode-se verificar que, de modo geral, não são consideradas as características e os interesses dos usuários para a apresentação adaptada do ambiente tridimensional, quanto ao conteúdo disponibilizado e à interface do ambiente, propriamente dita. Observa-se que a comunicação entre os agentes e usuários é feita, na maioria dos ambientes, através de diálogo em linguagem natural. Além disso, é apresentada, para grande parte dos ambientes, a interação entre os agentes e o usuário, sem a presença de outros agentes de software no ambiente. Por fim, nas referências não são encontradas informações acerca dos recursos aplicados nas manipulações de conteúdos nestes ambientes (quanto a inserção, remoção e atualização dos mesmos).

Tabela 5.1 – Comparativo entre os AVIs.

Sistema \ Requisito	Perfil Usuário	Interação Agente/Usuário	Ambiente	Avatar (usuário)	Agente	Tecnologia
Active Worlds	não	movimentação dos avatares	genérico, 3D, não adaptável, dinâmico	sim	repres. gráfica (povoamento do ambiente), vários	DirectX
Bersot et al (1998)	não	diálogo LN, auxílio a navegação	genérico, 3D, não adaptável, dinâmico	sim	repres. gráfica, conhec. ambiente, único	VRML
Rickel and Johnson (1997)	acompanhamento da interação	diálogo em LN, instruções uso equipam.	treinamento, 3D, dinâmico	não	repres. gráfica, reativo, cognitivo, único	VRML
Panayiotopoulos et al (1999)	não	consulta, auxílio a navegação	entretenimento, 3D/2D, não adaptável, estático	não	repres. gráfica, conhec. ambiente, único	VRML
Noll et al (1999)	coleta implícita (navegação, interação), individual	consulta, auxílio a navegação	entretenimento, 3D, estático	sim	repres. gráfica, conhec. ambiente, vários	Plataforma ASAP ¹⁸ , Java
Nijholt et al (2000)	não	diálogo em LN, auxílio navegação	entretenimento, 3D, não adaptável, estático	sim	repres. gráfica, conhec. ambiente, único	VRML
Milde (2000)	não	diálogo em LN, apresentação de produtos	<i>e-commerce</i> , 3D, não adaptável, estático	não	repres. gráfica, conhec. produtos, único	VRML, Java, Java3D, XML
Anastassakis et al (2001)	não	consulta	genérico, 3D/2D, não adaptável, estático	sim	repres. gráfica, reativo, cognitivo, vários	VRML, OpenGL,
Frery et al (2002)	coleta explícita (formulário), individual, estático	auxílio a navegação e sugestões	entretenimento, 3D/2D, adaptável (perfil), estático	não	repres. gráfica, conhec. do ambiente e usuário, único	VRML, Java
Rizzo et al (2002)	não	indicação de tarefas a realizar no ambiente	diagnóstico cognitivo, 3D, modificável, dinâmico	não	repres. gráfica, indicação de tarefas a realizar, vários	–
Chittaro and Ranon (2002)	coleta implícita e explícita (formulário e navegação), individual e grupo, dinâmico	consulta, auxílio a navegação	3D, <i>e-commerce</i> , adaptável (perfil), dinâmico	não	repres. gráfica, conhec. dos objetos, vários	VRML, Java
Chittaro et al (2003)	não	auxílio a navegação	3D, entretenimento, não adaptável, estático	não	repres. gráfica, conhec. ambiente, único	VRML, Java

¹⁸ A Simple Agent Platform para construção de ambientes virtuais colaborativos e agentes virtuais.

6 Ambiente Virtual Inteligente e Adaptativo

Nesta dissertação de mestrado é apresentado o ambiente AdapTIVE (*Adaptive Three-dimensional Intelligent and Virtual Environment*). Este ambiente tem a sua estrutura e apresentação adaptadas de acordo com os interesses e as preferências dos usuários (representadas em um modelo de usuário) e conforme a manipulação (inserção, remoção ou atualização) de conteúdos no ambiente. Um processo de categorização automático de conteúdos é aplicado na criação de modelos de conteúdos, usados na organização espacial dos mesmos no ambiente. No processo de adaptação, os modelos de usuário e conteúdo são utilizados. Além disso, um agente virtual inteligente atua como assistente dos usuários na navegação pelo ambiente e na localização de informações relevantes.

Neste capítulo o ambiente virtual inteligente e adaptativo proposto é detalhado. Inicialmente, na seção 1, é feita a contextualização do trabalho no estado da arte, onde são enfatizadas as contribuições do mesmo em relação aos assuntos e trabalhos revisados nos capítulos anteriores. Na seção 2, a arquitetura do ambiente é comentada e nas seções seguintes, cada componente da arquitetura é detalhado.

6.1 Contexto do Trabalho no Estado da Arte

Conforme Chittaro and Ranon (2002a), a capacidade de (semi)automaticamente adaptar o conteúdo, estrutura e/ou apresentação de um ambiente, de acordo com as características dos seus usuários é cada vez mais considerado um fator chave para incrementar o nível de satisfação dos mesmos. Em um ambiente tridimensional, a organização espacial das informações, de acordo com os interesses do usuário, pode facilitar a exploração do mesmo pelo usuário, contribuindo para uma interação mais efetiva. Conforme comentado no capítulo 2, as aplicações que provêm funcionalidades adaptativas, estão concentradas na aquisição e aplicação de modelos de usuários em ambientes bidimensionais e, de acordo com os trabalhos comentados no capítulo 5, o uso de modelos de usuários na adaptação de ambientes tridimensionais é ainda pouco explorado, mas promissor. Por fim, segundo Chittaro and Ranon (2000), diferentemente de ambientes bidimensionais, onde assume-se que os conteúdos das páginas acessadas pelo usuário foram vistos por ele, um ambiente tridimensional permite traçar melhor os

objetos que foram visualizados pelo usuário, através das seguintes verificações: (a) o usuário aproximou-se o suficiente de determinado objeto; (b) a câmera (que representa a direção da visão do usuário em relação ao espaço 3D) foi posicionada na direção do objeto.

Por outro lado, a organização espacial das informações em um ambiente tridimensional exige, em muitos casos, o agrupamento das mesmas, de acordo com algum critério semântico. Por exemplo, em uma loja virtual é interessante o agrupamento dos produtos conforme a seção a que pertencem (eletrônicos, bazar, outros) ou, em um ambiente de apoio a Educação a Distância (EaD), é necessário agrupar espacialmente os conteúdos conforme as áreas de conhecimentos a que pertencem. Neste contexto, modelos de conteúdos podem ser adotados na organização das informações no ambiente, sendo o especialista do domínio responsável pela definição dos modelos. No entanto, conforme comentado no capítulo 3, uma abordagem que pode ser utilizada na automatização da aquisição de modelos de conteúdos consiste na aplicação de técnicas de aprendizado de máquina na categorização de conteúdos textuais. Estas técnicas vêm sendo largamente aplicadas na organização e recuperação de informações, disponibilizadas em interfaces bidimensionais. No entanto, uma interessante aplicação para a categorização automática consiste na utilização deste processo na organização espacial de informações em AVs. Tal abordagem é ainda inexplorada em trabalhos relacionados aos AVIs.

Por fim, de acordo com Ballegooij and Eliëns (2001), os ambientes virtuais tridimensionais sofrem com o problema dos usuários sentirem-se “perdidos” no ambiente. Deste modo, recursos para prover assistência aos usuários na navegação pelo ambiente e localização de informações são necessários. Dentre os recursos geralmente utilizados, podem ser citados: a identificação de locais no ambiente (uso de setas e placas); os mapas 2D do ambiente; e os assistentes virtuais. Conforme Chittaro and Ranon (2003), a introdução de assistentes virtuais tem a vantagem adicional de tornar o ambiente mais natural e atrativo para o usuário. Assim, o uso de assistentes virtuais tem as seguintes vantagens: enriquece a interação com o ambiente virtual [Rickel et al, 2002]; torna o ambiente menos intimidador e mais natural ao usuário [Chittaro et al, 2003]; evita que os usuários sintam-se perdidos no ambiente [Rickel and Johnson, 2000]. Entretanto, considerando que em AVs é possível explorar situações que são impossíveis em um ambiente real, uma forma de conduzir o usuário até a informação

solicitada (sem a intervenção do agente) seria através do tele-transporte. No entanto, conforme Bowman et al (1998), o tele-transporte pode desorientar o usuário, especialmente nas primeiras interações, onde ele ainda não tem uma adequada representação mental do ambiente.

Conforme comentado no capítulo 5, o uso de agentes virtuais inteligentes vem sendo adotado em vários AVIs. No entanto, de um modo geral, tais agentes atuam em ambientes cuja estrutura se mantém inalterada no decorrer das interações com o usuário. Diferentemente destes agentes, o agente proposto atua em um ambiente dinâmico, atualizando o seu conhecimento a partir das alterações que ocorrem no ambiente, e estando apto a assistir o usuário, de forma transparente a estas alterações.

Deste modo, uma das contribuições importantes desta dissertação está em combinar, em um ambiente tridimensional, a aquisição e aplicação de modelos de usuários, a definição e utilização de modelos de conteúdos, e o uso de agentes inteligentes que atuam em ambientes dinâmicos. A organização destes componentes está refletida em uma arquitetura, AdapTIVE (*Adaptive Three-dimensional Intelligent and Virtual Environment*), definida considerando a possibilidade de sua utilização em aplicações onde a combinação destes componentes é promissora. Dentre as possíveis aplicações, são citadas as relacionadas a entretenimento (teatros virtuais), comércio eletrônico (lojas e shoppings virtuais) e educação (ambientes de apoio ao ensino). Na seção seguinte, a arquitetura AdapTIVE é detalhada.

6.2 Arquitetura AdapTIVE

O ambiente proposto consiste na representação de um mundo 3D, acessível através da Web, utilizado para a disponibilização de conteúdos. No ambiente, é dado suporte a dois tipos de usuários: requerentes da informação e provedores da informação. Os requerentes, representados por avatares, podem explorar o ambiente e solicitar auxílio ao agente virtual, para a navegação e localização de informações. Um modelo do requerente é mantido, de modo que o ambiente possa ser adaptado conforme os interesses e preferências do mesmo. Os provedores, responsáveis pelos conteúdos a serem disponibilizados, são auxiliados pelo agente na organização das informações, e podem explorar o ambiente. Os conteúdos são agrupados conforme as áreas a que

pertencem e possuem um modelo. Os modelos de conteúdos, de requerentes e de provedores são usados no processo de adaptação do ambiente.

A Figura 6.1 apresenta a arquitetura do ambiente. Conforme o *modelo do usuário (requerente ou provedor)* é feita a adaptação do ambiente. Esta adaptação envolve a personalização da apresentação e da estrutura do ambiente, onde os conteúdos de interesse do usuário são colocados, em uma ordem de visualização, mais próximos do que os conteúdos que não são de interesse. Este modelo contém informações sobre os interesses, preferências e comportamentos do usuário. Para a coleta dos dados utilizados na composição do modelo, são usadas as abordagens explícita e implícita, comentadas na seção 2.2. Na abordagem explícita, são aplicados questionários (para a coleta de dados, tais como nome e áreas de interesse), e na abordagem implícita são feitas observações da navegação no ambiente e verificação da interação com o agente (cujas informações serão registradas através do uso de *sensores*). A abordagem explícita é adotada para a aquisição de dados para composição de um modelo inicial, sendo a implícita aplicada para a atualização deste modelo.

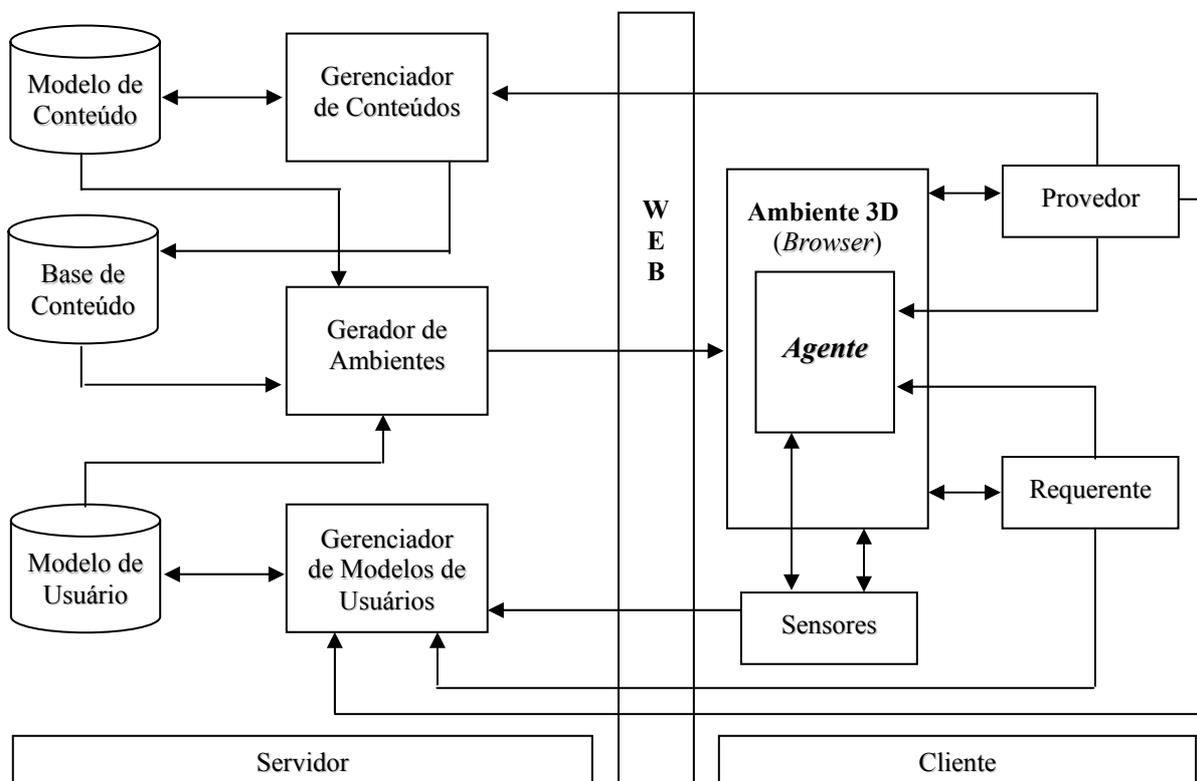


Figura 6.1 – Arquitetura AdapTIVE.

Um módulo, *gerenciador de modelos de usuários*, é o responsável pela inicialização e atualização dos modelos, a partir das informações repassadas pelo usuário e sensores, respectivamente.

Os conteúdos adicionados, removidos ou atualizados pelo provedor, são gerenciados pelo módulo *gerenciador de conteúdo* e mantidos em uma *base de conteúdos*. Cada conteúdo contém um modelo associado (*modelo de conteúdo*), com informações referentes a área ao qual pertence, tipo de mídia, palavras-chave que o caracterizam, entre outras. O provedor, auxiliado pelo processo de categorização automático de conteúdos textuais, atua na definição deste modelo.

A representação das informações no ambiente é feita através de componentes 3D, tais como objetos gráficos com hiperlinks para descrições mais detalhadas. Um módulo, *gerador de ambientes*, é o responsável pela geração das estruturas 3D que formam o ambiente, permitindo a construção de ambientes adaptados, de acordo com os modelos de usuário e de conteúdo. Além disso, este módulo repassa ao agente as informações referentes aos modelos dos usuários que estão interagindo com o ambiente e informações sobre os conteúdos e suas localizações, de forma que ele possua informações suficientes para o auxílio aos usuários.

Nas seções que seguem, são detalhados os principais componentes da arquitetura: gerenciador de modelos de usuários, gerenciador de conteúdos, gerador de ambientes, e agente virtual inteligente. Neste capítulo é dada ênfase à descrição destes componentes, sendo os mesmos novamente abordados no capítulo 7, onde é dada uma visão prática da arquitetura, com exemplos relacionados ao protótipo implementado.

6.2.1 Gerenciador de Modelos de Usuários

O gerenciador de modelos de usuários é o módulo responsável pela inicialização e atualização dos modelos dos usuários. O modelo do usuário contém informações sobre os interesses, preferências e comportamentos do usuário. Em relação as propriedades do modelo (conforme as propriedades comentadas na seção 2.1), sua especialização é individual (refletindo as propriedades de um único usuário); o modelo é dinâmico (podendo ser alterado durante o curso da interação); sua extensão temporal é de curto e longo prazo (as evidências mais recentes e as passadas são consideradas na formação e

atualização do modelo); e o método de uso é descritivo (está descrito em uma base de dados).

Para a coleta dos dados usados na composição do modelo, as abordagens implícita e explícita são utilizadas. A abordagem explícita é adotada na aquisição dos dados para composição de um modelo inicial do usuário, sendo a implícita aplicada para atualizar este modelo. Na abordagem explícita, um formulário é usado para coletar os seguintes dados: nome, e-mail, gênero, áreas de interesse e preferência por cores. As três últimas informações são utilizadas na adaptação inicial do ambiente. Na abordagem implícita, são realizadas a monitoração da navegação do usuário no ambiente e a verificação das interações com o agente.

O processo de atualização do modelo do usuário é baseado no emprego de regras e fatores de certeza (FC) ([Nikolopoulos, 1997]; [Giarratano and Riley, 1998]). As regras, formalismo comumente adotado em sistemas especialistas para a representação do conhecimento, permitem inferir conclusões (hipóteses) com base em antecedentes (evidências). Para cada hipótese, é possível associar um FC, o qual representa o grau de crença associado a hipótese. As regras são descritas no seguinte formato:

$$\begin{array}{l} \text{SE Evidência(s)} \\ \text{ENTÃO Hipótese com FC} = x \end{array} \quad (6.1)$$

Os FCs associam medidas de crença (MC) e descrença (MD) a uma hipótese (H), dada uma evidência (E). Um FC 1 indica crença total em uma hipótese, enquanto -1 corresponde a descrença total. O cálculo do FC é realizado a partir das equações (6.2), (6.3) e (6.4) (onde P(H) correspondente a probabilidade da hipótese H ser verdadeira e P(H|E) correspondente a probabilidade da hipótese H ser verdadeira, dada a evidência E).

$$FC = \frac{MC - MD}{1 - \text{MIN}(MC, MD)} \quad (6.2)$$

$$MC \begin{cases} 1 & \text{if } P(H) = 1 \\ \frac{\text{MAX}[P(H|E), P(H)] - P(H)}{1 - P(H)} & \text{caso contrário} \end{cases} \quad (6.3)$$

$$MD \begin{cases} 1 & \text{if } P(H) = 0 \\ \frac{\text{MIN}[P(H|E), P(H)] - P(H)}{0 - P(H)} & \text{caso contrário} \end{cases} \quad (6.4)$$

Nesta dissertação, as evidências estão relacionadas com os locais visitados no ambiente e aos conteúdos solicitados e acessados pelo usuário. As hipóteses, inferidas com base nas evidências, correspondem aos interesses do usuário pelas áreas correspondentes.

O interesse inicial do usuário em uma dada área – valor inicial para $P(H)$ – é determinado pela coleta explícita. Nesta coleta, o usuário indica: interesse pela área ($P(H) = 1$); indiferença quanto aos conteúdos da área ($P(H) = 0.5$); ou desinteresse pela área ($P(H) = 0$). Os valores iniciais das $P(H)$ s são utilizados na organização inicial do ambiente, permitindo que os conteúdos das áreas de interesse sejam colocados, em uma ordem de visualização, mais próximos do que os conteúdos que não são de interesse.

No decorrer da interação do usuário com o sistema, as evidências são coletadas e, conforme uma janela de tempo para a atualização do modelo inicial (n sessões), os FCs, para cada área são mensurados. Para a inferência das hipóteses de interesse, as regras (6.5), (6.6), (6.7), (6.8) foram definidas.

As regras (6.5), (6.6) e (6.7) são utilizadas quando as evidências de solicitação, navegação e/ou acesso existem. Neste caso, é feita a combinação das regras correspondentes as evidências existentes e o FC resultante é calculado conforme a equação (6.9). A regra (6.8) é especialmente usada quando não existem evidências, o que é considerado um desinteresse do usuário pela área correspondente.

$$\begin{array}{l} \text{SE solicitou} \\ \text{ENTÃO interesse em Y com FC} = \mathbf{x} \end{array} \quad (6.5)$$

$$\begin{array}{l} \text{SE navegou} \\ \text{ENTÃO interesse em Y com FC} = \mathbf{x} \end{array} \quad (6.6)$$

$$\begin{array}{l} \text{SE acessou} \\ \text{ENTÃO interesse em Y com FC} = \mathbf{x} \end{array} \quad (6.7)$$

$$\begin{array}{l} \text{SE (não solicitou) E (não navegou) E (não acessou)} \\ \text{ENTÃO interesse em Y com FC} = \mathbf{x} \text{ (onde } \mathbf{x} < 0 \text{)} \end{array} \quad (6.8)$$

Regras combinadas como segue (2 regras com FC1 e FC2):

$$FC = \begin{cases} FC1 + FC2 (1 - FC1) & \text{Se ambos } > 0 \\ \frac{FC1 + FC2}{1 - \text{MIN}(|FC1|, |FC2|)} & \text{Se um } FC < 0 \\ FC1 + FC2 (1 + FC1) & \text{Se ambos } < 0 \end{cases} \quad (6.9)$$

Deste modo, a cada n sessões (janela de tempo ajustável), para cada área, as evidências (solicitação, navegação e acesso) são verificadas, as inferências nas regras são feitas e os FCs correspondentes as hipóteses de interesse são mensurados. A partir da ordenação dos FCs resultantes, é possível estabelecer um ranking das áreas de interesse do usuário e verificar as alterações no modelo inicial, de forma a atualizá-lo. A partir da atualização, a re-organização do ambiente pode ser feita. Entretanto, simplesmente alterar a organização do ambiente pode desorientar o usuário. Portanto, a re-organização é sugerida ao usuário e autorizada pelo mesmo.

Cabe destacar, entretanto, que foram adicionados os seguintes parâmetros ao processo de atualização do modelo: número de sessões para a revisão do modelo (janela de tempo para revisão), peso associado à evidência de acesso, persistência do acesso, limiar para aumento da $P(H)$ e limiar para diminuição da $P(H)$. A seguir, são comentadas as justificativas para o uso destes parâmetros.

O número de sessões para a revisão corresponde a janela de tempo utilizada para a revisão do modelo. Deste modo, pode-se configurar o intervalo de sessões a partir do qual o modelo deve ser revisado. Por exemplo, com um intervalo igual a 7, a cada 7 sessões são coletadas as evidências, inferidas as hipóteses de interesse, re-estabelecido o ranking das áreas e feita a re-organização do ambiente (caso o usuário aceite a re-organização).

Além do uso de uma janela de tempo para a revisão, o acesso a um determinado conteúdo (clique) pode ser considerado uma forte evidência de interesse. Deste modo, deve-se garantir que a medida de crença (MC) seja superior a zero e a medida de descrença (MD) igual 0. Para isto, associa-se um peso à evidência de acesso, tal que $P(H|A) = P(H) + \text{peso}$. Além disso, o uso de uma persistência para o acesso (número de

sessões, a partir do último acesso, em que o peso deve ser considerado) permite que o interesse pela dada área, em relação a esta evidência, seja atenuado ao longo das sessões que seguem o último acesso. Através do ajuste do peso e da persistência é possível determinar o grau de relevância da evidência de acesso, na quantificação do interesse do usuário pela área correspondente (o grau de interesse tende a aumentar com o uso de pesos maiores).

Os parâmetros seguintes, limiares para aumento e diminuição da $P(H)$, são utilizados para que os seus valores iniciais sejam atualizados conforme as evidências coletadas. Isto se faz necessário porque, por exemplo, caso o usuário tenha informado (na coleta explícita) desinteresse inicial ($P(H) = 0$) por determinada área, mas existem evidências de solicitação, navegação e/ou acesso, o FC correspondente será -1, indicando total descrença do interesse do usuário pela área (considerando que $MD = 1$ se $P(H) = 0$). Deste modo, o ajuste dos valores iniciais das $P(H)$ s se faz necessário. Para isto, a cada sessão, são verificados os aumentos das medidas de crença (MC) e descrença (MB). Caso as variações destas medidas atinjam um limiar pré-determinado, o valor da $P(H)$ é alterado. O limiar para aumento da $P(H)$ é utilizado na verificação da variação da MC, enquanto o limiar para diminuição é aplicado na verificação da variação da MD. O valor da $P(H)$ é alterado, conforme a faixa de valores 0, 0.5 e 1. Retomando o exemplo acima, se a $P(H)$ inicial é 0, mas existem evidências, quando a MC atingir o limiar pré-determinado de aumento da $P(H)$, o valor correspondente será alterado para $P(H) = 0.5$.

Cabe destacar, entretanto, que a abordagem proposta para a atualização de modelos de usuários, apresentada nesta seção, é novamente comentada no capítulo 7, onde será dada uma visão prática de sua aplicação.

6.2.2 Gerenciador de Conteúdos

O módulo gerenciador de conteúdo é o responsável pelas inserções, remoções e atualizações de conteúdos no ambiente, e pelo gerenciamento dos modelos correspondentes. O provedor é o responsável pela definição dos modelos, podendo ser auxiliado pelo processo categorização automático, especialmente para conteúdos textuais. Na inserção de um conteúdo, o provedor informa os dados para a composição do modelo através de um formulário. Dentre os dados utilizados, pode-se citar:

categoria do conteúdo (dentre um conjunto pré-determinado), título, descrição, palavras-chave, tipo de mídia e arquivo correspondente. Para conteúdos textuais, o processo de categorização automático (invocado através do formulário), pode ser utilizado na extração da categoria do conteúdo e do conjunto de palavras-chave que o caracteriza.

O processo de categorização automático proposto, conforme comentado no capítulo 3, está baseado na aplicação de técnicas de aprendizado de máquina. Os resultados obtidos com os experimentos preliminares (seção 3.5), indicaram uma melhor performance das Árvores de Decisão, sendo, então, adotado o algoritmo de aprendizado C4.5 no processo de categorização associado ao gerenciador de conteúdos.

Considerando que na aplicação deste tipo de técnica, é exigida a coleta prévia de um conjunto de exemplos para o aprendizado e validação do algoritmo, as áreas a serem contempladas no ambiente devem ser definidas e os exemplos correspondentes coletados. Após, deve ser realizado o pré-processamento dos exemplos e geração dos scripts a serem submetidos à ferramenta C4.5, para a geração do categorizador. Estas atividades são suportadas por uma aplicação em Java, comentada na seção 3.4. Após, o “modelo aprendido” – regras extraídas da árvore – é conectado ao módulo gerenciador de conteúdo, de forma que seja utilizado na categorização de novos conteúdos. Para isto, uma aplicação em Java¹⁹, converte as regras extraídas da árvore em regras do tipo IF – THEN. Esta aplicação recebe como entrada um arquivo *tree* (arquivo de saída da ferramenta C4.5, contendo a descrição da árvore gerada) e gera um arquivo *java* com as regras IF – THEN correspondentes. Desta forma, quando um novo conteúdo textual é submetido ao processo de categorização automático, ele é pré-processado, categorizado e tem suas palavras-chave extraídas.

6.2.3 Gerador de Ambientes

O módulo gerador de ambiente é o responsável pela criação dos ambientes, conforme os modelos de usuários e conteúdos. A partir da verificação das áreas de interesse dos usuários e dos conteúdos disponíveis para as mesmas, são criados os ambientes adaptados. Na adaptação, os conteúdos correspondentes as áreas de maior interesse, são colocados, em uma ordem de visualização, mais próximos dos conteúdos correspondentes as áreas que não são de interesse.

¹⁹ Disponível em <http://www.inf.unisinos.br/~cassiats/dissertacao/>

Neste processo, uma aplicação em Java gera arquivos *.wrl* (VRML), contendo a estrutura 3D básica do ambiente (paredes, chão e teto), considerando as preferências por cores, definidas no modelo do usuário. Os arquivos *.wrl* são carregados por uma segunda aplicação, responsável por gerar os objetos 3D que representam os conteúdos no ambiente e distribuí-los espacialmente, conforme os seus modelos. Os objetos 3D que representam os conteúdos no ambiente podem conter pequenas descrições sobre os mesmos, especialmente nos casos dos conteúdos textuais. O usuário, a partir do clique na descrição do conteúdo pode, então, visualizar o conteúdo correspondente.

Além disso, este módulo repassa ao agente inteligente as informações pertinentes aos modelos dos usuários que estão interagindo com o ambiente e informações sobre os conteúdos e suas localizações, de forma que ele possua informações suficientes para o auxílio aos usuários.

6.2.4 Agente Virtual Inteligente

A função do agente inteligente no ambiente é assistir os usuários na navegação e localização de informações relevantes. Ele possui as seguintes características: percepção, habilidade para interagir, conhecimento, certo grau de raciocínio e reatividade, e representação gráfica. A percepção representa as observações do agente durante a interação com o usuário, e sua habilidade de interação está relacionada com a troca de informações com o usuário. O conhecimento do agente representa as informações que ele possui sobre o ambiente, o qual pode ser atualizado durante a interação, conforme as alterações que ocorrem no ambiente. Além disso, ele possui certo grau de raciocínio e reatividade, o que lhe confere a capacidade de agir no ambiente, conforme a análise das solicitações do usuário.

A partir das classificações atribuídas aos agentes inteligentes, apresentadas na seção 4.3, o agente proposto pode ser classificado como sendo: híbrido (com comportamentos reativos e cognitivos, conforme detalhado a seguir); informativo (oferece suporte ao usuário na recuperação de informações); de interface (interage diretamente com o usuário); virtual (atua em um ambiente virtual); e com tipo de atuação isolada (no que tange a interação com outros agentes computacionais).

Para a especificação de como o agente se comporta durante a interação com o ambiente e na realização de suas tarefas, foi selecionada a arquitetura de controle híbrida, (comentada na seção 4.4.3). A Figura 6.2 ilustra a arquitetura do agente.

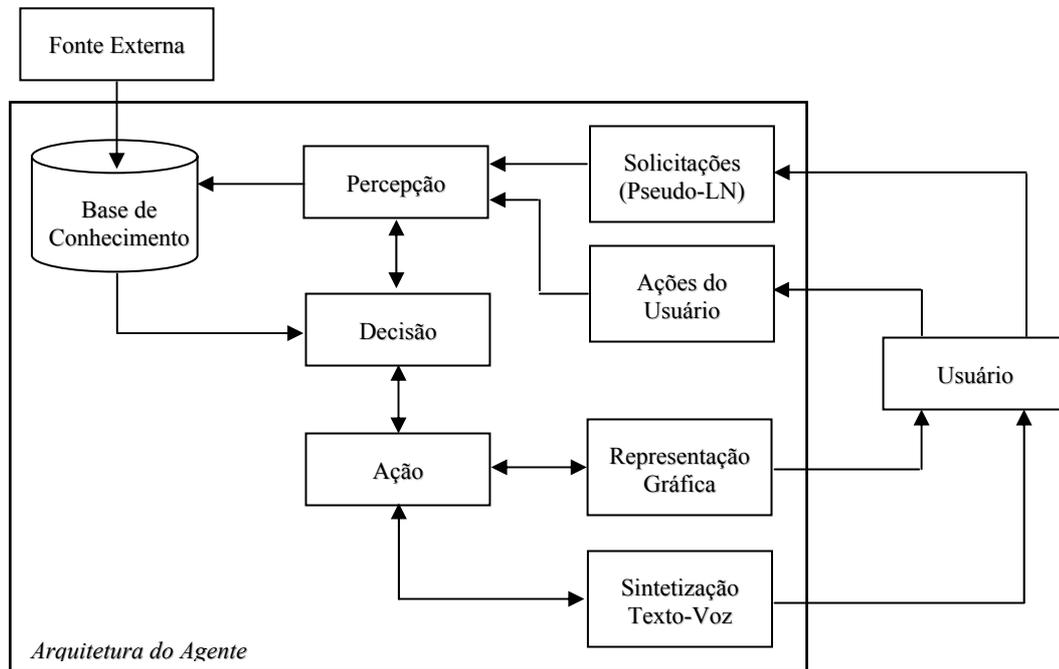


Figura 6.2 – Arquitetura do agente virtual inteligente.

O conhecimento do agente, armazenado em uma *base de conhecimento*, contempla, principalmente, informações acerca do ambiente, envolvendo as informações sobre os conteúdos disponibilizados e suas localizações. Tais informações são advindas de uma *fonte externa*, que representa o módulo gerador de ambientes. Além disso, um mapa topológico do ambiente é mantido na base de conhecimento do agente, contendo um conjunto de rotas para posições-chave do ambiente é armazenado. De acordo com o mapa topológico e a informação que o agente possui do ambiente, o agente define o conjunto de rotas que deve ser seguido na localização de determinado conteúdo ou na navegação de sua posição atual até determinado local do ambiente. Considerando que o agente atualiza o seu conhecimento a cada modificação no ambiente, ele está sempre apto a verificar o conjunto de rotas que leva às novas posições dos conteúdos.

O *módulo de percepção* atua na observação da interação com os usuários, sendo as informações obtidas desta interação armazenadas na base de conhecimento do agente, para posterior utilização na atualização do modelo do usuário. É através deste módulo que o agente detecta as *solicitações* do usuário para auxílio à navegação e à localização

de informações, as quais são feitas a partir de um diálogo em pseudo Linguagem Natural (LN), comentado a seguir; e observa as *ações do usuário* no ambiente (movimentação pelo ambiente).

A partir de sua percepção e do conhecimento que possui, o agente decide como agir no ambiente. O *módulo de decisão* é o responsável por esta atividade. As decisões tomadas são repassadas ao *módulo de ação*, responsável pela execução das decisões indicadas pelo módulo de decisão, a partir da manipulação da interface gráfica do agente e da sintetização de texto-voz de mensagens (comentadas a seguir), apresentadas ao usuário. Um sintetizador permite que as pequenas sentenças, também apresentadas em uma interface de comunicação textual, sejam convertidas para voz.

Deste modo, o agente possui dois tipos de comportamento: reativo e cognitivo. A partir de um comportamento cognitivo, o agente define quais as ações devem ser executadas para que os seus objetivos (navegação pelo ambiente ou localização de informações) sejam alcançados. Com um comportamento reativo, a partir da sua percepção, são geradas as ações, as quais envolvem a manipulação de sua representação gráfica (movimentação do corpo) e a sintetização texto-voz.

Em relação a comunicação entre o agente e os usuário, esta é realizada de um modo verbal, através de uma linguagem pseudo-natural, e de forma não verbal, através das ações do agente no ambiente. O diálogo em linguagem pseudo-natural é formado por um grupo pré-definido de perguntas e respostas, formadas por um verbo que corresponde ao tipo de requisição do usuário (auxílio a navegação ou localização de informações) e um complemento, indicando o objeto de interesse do usuário.

Durante a requisição de ajuda para localizar uma informação, por exemplo, o usuário pode indicar *Localizar<conteúdo>*; e para solicitar auxílio à navegação, o usuário pode indicar *Navegar<área>*. As respostas do agente são indicadas através de sua própria movimentação pelo ambiente, por indicações através de pequenas sentenças e por sintetização de texto-voz. Especialmente na interação com o provedor, durante a inserção de um conteúdo, ele pode indicar *Inserir <conteúdo>*, e o agente apresentará a interface para entrada dos dados para a especificação e identificação do modelo do conteúdo. Através desta interface, o processo de categorização pode ser invocado. Entretanto, como uma forma de simplificar o modelo de comunicação, algumas opções

pré-definidas estão disponíveis ao usuário, através de um mecanismo de seleção de opções.

Neste capítulo foi apresentada a arquitetura AdapTIVE, e dada uma visão alto nível de seus componentes principais. Como forma de validá-la, foi implementado um protótipo, que compreende um ambiente para disponibilização de conteúdos, no contexto da Educação a Distância (EaD). No capítulo que segue, o protótipo é apresentado.

7 Protótipo

Para validar a arquitetura proposta, foi desenvolvido um protótipo de um ambiente de apoio a Educação a Distância (EaD). O ambiente é utilizado para a disponibilização de conteúdos educacionais. A motivação para o desenvolvimento deste ambiente fundamentou-se nas seguintes premissas: dinamicidade dos ambientes de apoio a EaD (atualizações contínuas de conteúdos), diversidade de modelos de usuários; e uso promissor dos AVIs, permitindo a criação de ambientes altamente interativos, motivando o estudante e enriquecendo o seu aprendizado.

Conforme Avradinis et al (2000), nos ambientes virtuais tridimensionais, o usuário é uma parte do sistema, uma presença autônoma no ambiente. Ele está livre para navegar, interagir com objetos e examinar o ambiente de diferentes pontos de vista, o que não é possível em ambientes bidimensionais. Além disso, segundo Frery et al (2002), o paradigma 3D é útil, principalmente, por oferecer a possibilidade de representar a informação de um modo realístico, organizando o conteúdo de uma maneira espacial. Através da utilização de componentes 3D, obtém-se maior intuição na visualização da informação, permitindo ao usuário explorá-la de um modo interativo, mais natural ao ser humano. Estas características tornam o uso de ambientes virtuais tridimensionais promissores em EaD. Além disso, em ambientes virtuais educacionais é desejável que o usuário sinta-se “imerso” no ambiente. Os resultados apresentados por Nunez and Blake (2003) indicam que os ambientes virtuais gráficos produzem estatisticamente maiores níveis de presença do que os baseados em texto.

Cabe destacar, entretanto, que a arquitetura proposta pode ser estendida para aplicações onde a combinação entre modelos de usuário e conteúdo no processo de adaptação do ambiente seja promissora. Dentre as possíveis aplicações, as relacionadas a comércio eletrônico podem ser citadas. Por exemplo, shoppings virtuais tridimensionais oferecem ao usuário uma interação mais natural e atrativa, considerando que ele está apto a navegar através da loja virtual e visualizar produtos em um modo mais realístico, se comparado com interfaces bidimensionais. Neste contexto, a possibilidade de organizar as lojas e seus produtos, de acordo com as necessidades e

preferências de cada um dos usuários (impraticável em um ambiente real), facilita o acesso aos produtos e incrementa o nível de satisfação dos usuários. Além disso, um assistente inteligente que estabelece um diálogo com o usuário e oferece auxílio para a localização de lojas ou seus produtos, ou oferece informações sobre os mesmos, pode tornar a interação ainda mais interessante.

Neste capítulo, o protótipo desenvolvido é detalhado, sendo explorados os principais componentes da arquitetura AdapTIVE. Na seção 1, as áreas contempladas no ambiente e a estrutura do mesmo são apresentadas. Na seção 2, o gerenciador de modelos de usuários é comentado. A seção 3 aborda o processo de modelagem de conteúdos. Na seção 4, o agente virtual é apresentado. Por fim, na seção 5, uma extensão da estrutura inicial do ambiente é comentada.

7.1 Estrutura do Ambiente

No protótipo, desenvolvido em Java3D²⁰, VRML²¹ e MySQL²², foi adotada uma divisão do ambiente virtual conforme as áreas do conhecimento dos conteúdos educacionais disponibilizados. A cada área estão associadas sub-áreas, representadas como sub-ambientes. As seguintes áreas e sub-áreas foram selecionadas para o protótipo: Inteligência Artificial (IA) – Redes Neurais Artificiais, Algoritmos Genéticos e Sistemas Multiagentes; Computação Gráfica (CG) – Modelagem, Animação e Visualização; Redes de Computadores (RC) – Segurança, Gerência e Protocolos; Engenharia de Software (ES) – Análise e Projeto de Sistemas, Padrões e Qualidade de Software. A cada área está associada uma sala no ambiente e as sub-áreas estão representadas como sub-salas. As Figuras 7.1 e 7.2 apresentam interfaces do protótipo que ilustram a divisão do ambiente em salas e sub-salas. Em cada sub-sala, são dispostos os conteúdos correspondentes a sub-área, e o usuário deve percorrer as salas das áreas para atingir as sub-salas. A interação entre o usuário e o agente é realizada através de uma interface textual de comunicação (similar a de *chat*), através da qual o usuário seleciona o tipo de solicitação a ser emitida ao agente, e o mesmo lista as suas respostas. Além disso, como forma de facilitar a localização do usuário no ambiente, um mapa 2D do ambiente é disponibilizado ao usuário.

²⁰ API disponível em: <http://java.sun.com/products/java-media/3D>

²¹ VRML – *Virtual Reality Modeling Language*, padronizada pelo www.web3d.org

²² MySQL – *Open Source Database*, disponível em www.mysql.com

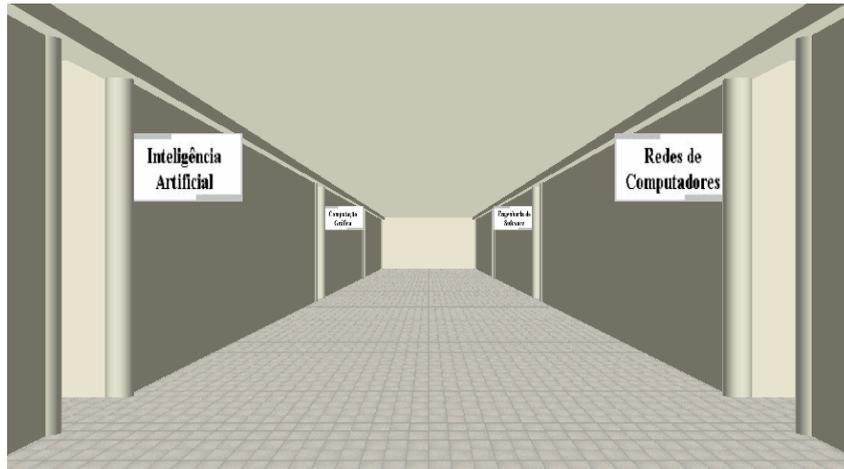


Figura 7.1 – Salas do ambiente.

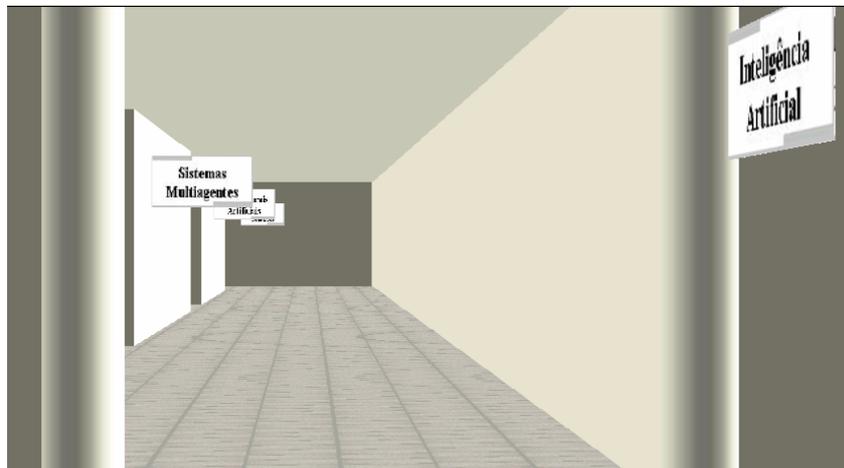


Figura 7.2 – Sub-salas do ambiente.

O módulo *Gerador de Ambientes* (comentado na seção 6.2.3) é o responsável pela geração dos arquivos *.wrl* (VRML), com diferentes configurações. Ambientes com diferentes números de salas (vazias ou com sub-salas, por exemplo) são gerados. Os arquivos *wrl* são gerados por uma aplicação em Java, e posteriormente carregados por uma segunda aplicação, desenvolvida com a API Java3D, utilizando o pacote *Vrml97*²³. As Figuras 7.3 (a) e 7.4 (a) representam exemplos do ambiente com diferentes configurações. As Figuras 7.3 (b) e 7.4 (b) ilustram os mapas 2D correspondentes a organização dos ambientes 3D.

²³ Especificação disponível em: <http://www.web3d.org/technicalinfo/specifications/vrml97/>

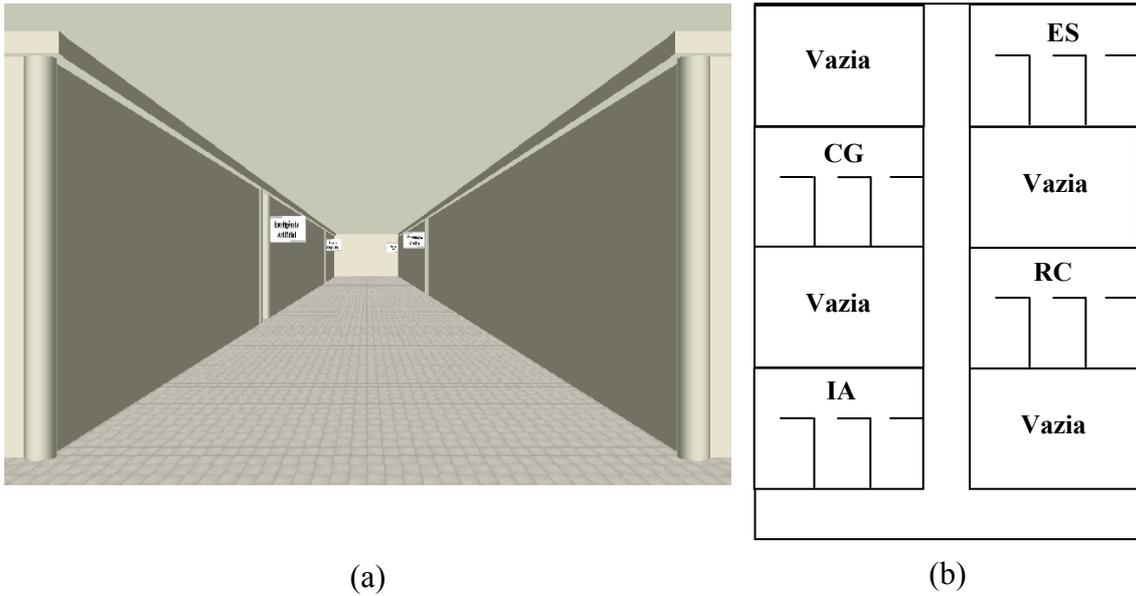


Figura 7.3 – (a) Ambiente 3D com salas, vazias e com sub-salas; (b) Mapa 2D correspondente.

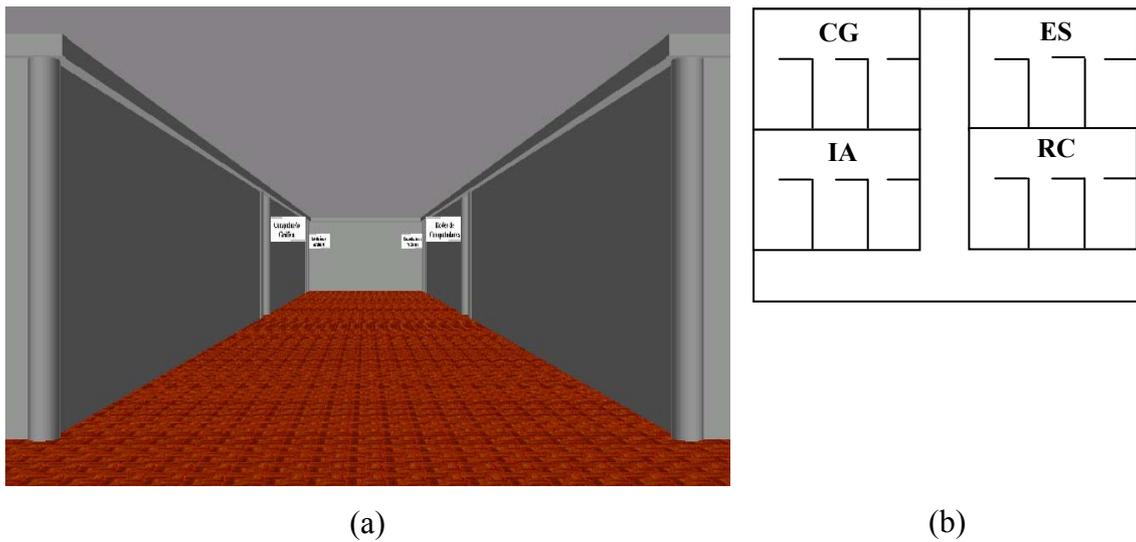


Figura 7.4 – (a) Ambientes com salas com sub-salas; (b) Mapa 2D correspondente.

7.2 Gerenciador de Modelos de Usuário

De acordo com o modelo do usuário, é feita a organização do ambiente: as salas que correspondem as áreas de maior interesse do usuário são colocadas, em uma ordem de visualização, mais próximas das salas cujos conteúdos não são de interesse. Deste modo, não são eliminados do ambiente os conteúdos que não são de interesse do usuário, mas sim enfatizados os de interesse, permitindo que o usuário tenha a liberdade para visualizar conteúdos diversos.

Os dados para a composição do modelo inicial do usuário são obtidos a partir de um formulário (coleta explícita). Neste formulário, apresentado na Figura 7.5, o usuário indica os seus dados gerais, suas preferências por cores e suas áreas de interesse.

AdapTIVE - Adaptive, Three-Dimensional, Intelligent and Virtual Environment

Formulário para Cadastro de Usuário

Dados Gerais

Nome:

Gênero:

E-Mail:

Login:

Senha:

Confirma Senha:

Preferências

Cores:

Áreas do Conhecimento

	Interesse		
	Sim	Não	Indiferente
Computação Gráfica	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Engenharia de Software	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Inteligência Artificial	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Redes de Computadores	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Confirmar Limpar

Figura 7.5 – Formulário para a coleta dos dados para composição do modelo inicial.

Conforme comentado na seção 6.2.1, para cada área são definidas as probabilidades de interesse ($P(H)$). Para as áreas nas quais o usuário tem interesse, a probabilidade de interesse é ajustada para 1 ($P(H) = 1$); para as áreas em que o usuário é indiferente, é atribuído o valor 0.5 à probabilidade ($P(H) = 0.5$); e para as áreas em que o usuário não apresenta interesse, o valor da probabilidade é ajustado para 0 ($P(H) = 0$). A partir do ranking das probabilidades iniciais, é estabelecida a organização inicial do ambiente. As Figuras 7.6 e 7.7 ilustram exemplos de adaptações iniciais do ambiente. No ambiente da Figura 7.6, o usuário (gênero feminino) tem interesse por IA e preferência por cores claras; no ambiente da Figura 7.7, o usuário (gênero masculino), tem interesse por CG e preferência por cores escuras.



Figura 7.6 – Usuário com interesse em IA.

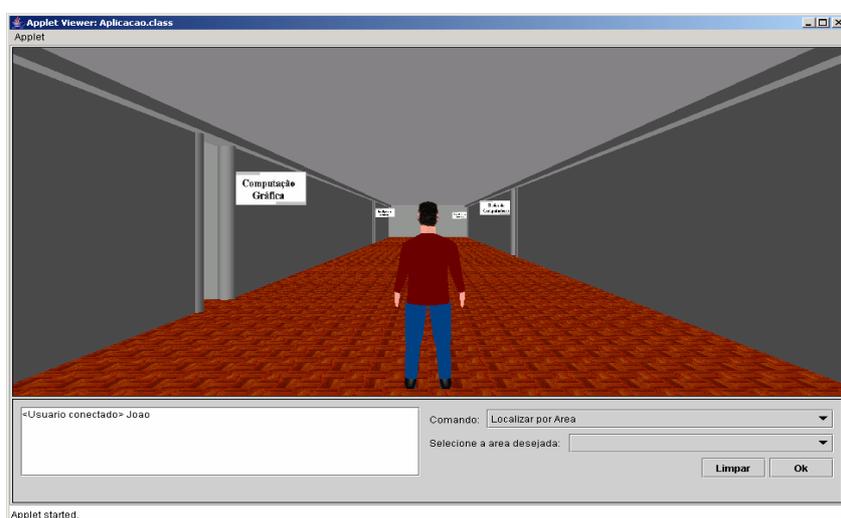


Figura 7.7 – Usuário com interesse em CG.

À medida que o usuário interage com o ambiente, as evidências de navegação, solicitação e acesso aos conteúdos são coletadas e utilizadas no processo de atualização do modelo do usuário. Após n sessões (janela de tempo ajustável), para cada área, conforme as evidências coletadas, as regras (indicadas na seção 6.2.1) são aplicadas e calculados os FCs correspondentes. A partir da ordenação dos FCs, é feito um novo ranking das áreas de interesse do usuário e, deste modo, é feita a re-organização do ambiente. É importante ressaltar que as alterações no ambiente são sugeridas ao usuário e apenas realizadas se autorizadas pelo mesmo.

Por exemplo, considerando um usuário com interesse em IA ($P(H) = 1$), indiferente quanto a CG ($P(H) = 0.5$) e RC ($P(H) = 0.5$) e que não apresenta interesse inicial em ES ($P(H) = 0$), o ranking inicial das áreas, em ordem decrescente de interesse,

seria composto por IA, CG, RC e ES, conforme as $P(H)$ s correspondentes. Dadas as evidências coletadas a partir do acompanhamento do uso do sistema pelo usuário, apresentadas no gráfico da Figura 7.8, o ranking inicial, ao final da sétima sessão (janela de tempo considerada), deve ser reavaliado.

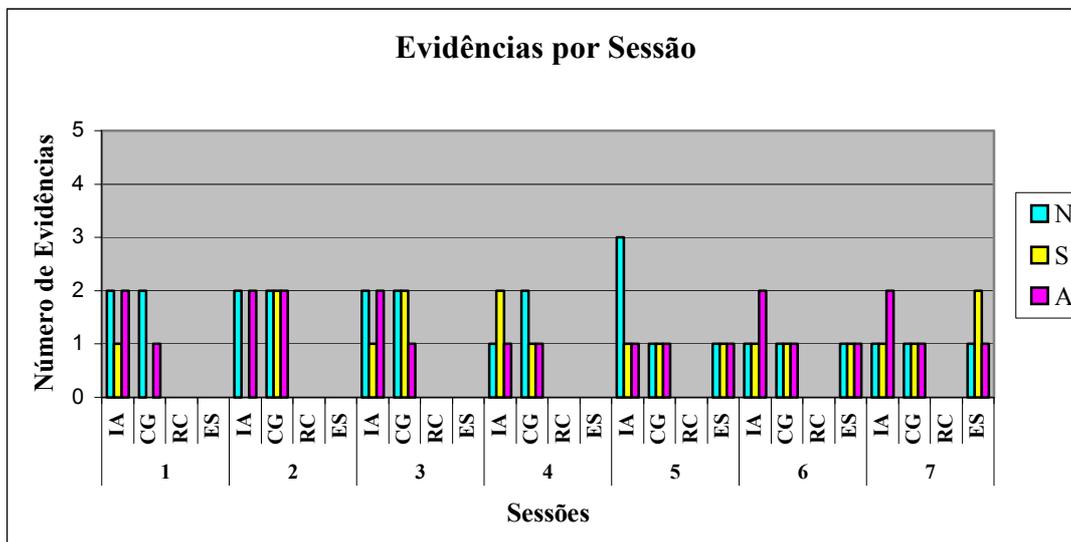


Figura 7.8 – Número de solicitações (S), navegações (N) e acesso (A).

De acordo com a Figura 7.8, é verificado que a área de RC não foi solicitada, navegada ou teve seus conteúdos acessados e, por outro lado, o usuário passou a navegar, solicitar e acessar a área de ES. Conforme apresentado no gráfico da Figura 7.9, um aumento do FC, correspondente a área de ES, pode ser observado. Desta forma, ao final da sétima sessão, os FCs resultantes seriam: 1, -1, 0.4 e 0.2 (IA, RC, CG, ES, respectivamente). Através da ordenação dos FCs resultantes, é possível detectar uma alteração no modelo inicial do usuário, onde o novo ranking das áreas de interesse seria: IA, CG, ES e RC.

Para o exemplo apresentado foram utilizados os seguintes valores para os parâmetros (comentados na seção 6.2.1): peso da evidência de acesso igual a 0.2; persistência de acesso com valor de 3 sessões; e limiares para aumento e diminuição da $P(H)$ com valores de 0.1 e 0.3, respectivamente. Estes valores foram definidos de forma empírica e intuitiva, considerando o contexto didático do exemplo. Cabe destacar, entretanto, que um conjunto de experimentos iniciais foi realizado, de forma a verificar o impacto da variação dos parâmetros no processo de atualização de um modelo de usuário.

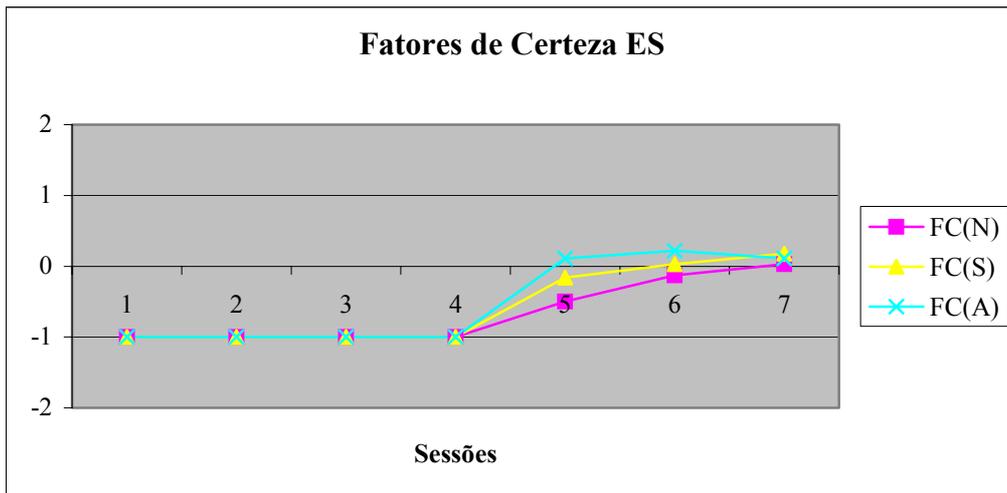


Figura 7.9 – FCs correspondentes a área de ES, ao longo das 7 sessões.

As Figuras 7.10 (a) e 7.10 (b) representam exemplos esquemáticos da organização do ambiente antes e após a modificação no modelo do usuário, conforme o exemplo comentado acima. Cabe destacar que as salas vazias no ambiente foram adicionadas propositalmente, de forma a enfatizar as alterações das localizações das salas no ambiente. As Figuras 7.11 (a) e 7.11 (b) ilustram as interfaces do ambientes, correspondentes ao esquema apresentado nas Figuras 7.10 (a) e 7.10 (b).

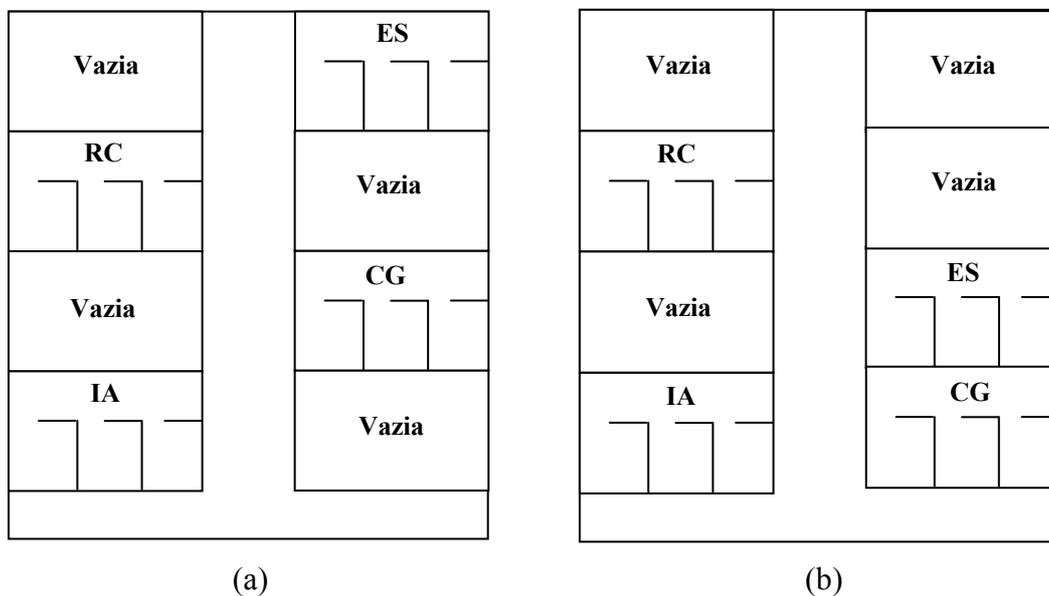


Figura 7.10 – (a) Organização do ambiente de acordo com o modelo inicial do usuário; (b) organização do ambiente após alterações no modelo do usuário.

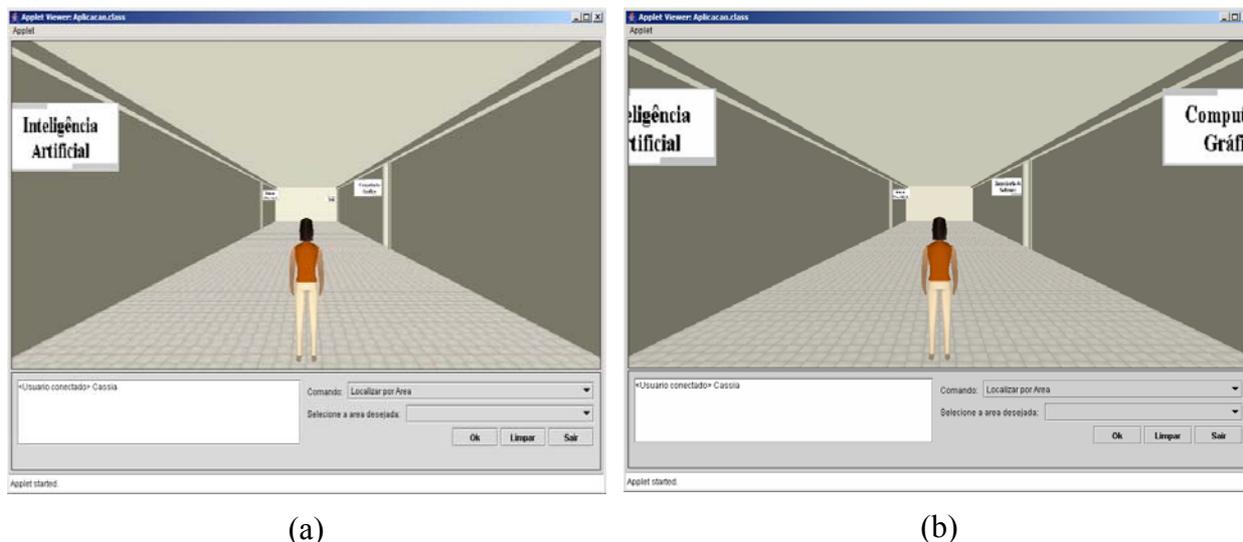


Figura 7.11 – (a) Organização do ambiente de acordo com o modelo inicial do usuário;
 (b) Organização do ambiente após alterações no modelo do usuário.

7.3 Gerenciador de Conteúdos

No ambiente, os seguintes tipos de conteúdos são suportados: *.txt , *.html, *.doc, *.pdf, *.ppt, *.jpg, *.bmp, *.wrl (VRML), *.avi, *.wav e *.au. Os tipos que correspondem a imagens 2D e 3D (.bmp, .jpg e .wrl) são representados diretamente ao ambiente, sendo os demais representados por objetos gráficos com pequenas descrições e *links* para os arquivos correspondentes (para visualização no respectivo editor). Além disso, os arquivos de som podem ser ativados quando o usuário navega por determinada área ou clica em determinado objeto.

A Figura 7.12 apresenta um exemplo da representação de conteúdos no ambiente, para a sub-área Redes Neurais Artificiais (RNAs). Na figura, uma simplificação de uma RNA é apresentada através de um objeto .wrl; uma imagem .jpg ilustra a organização de uma RNA do tipo SOM (*Self-Organizing Maps*); e uma pequena descrição de um conteúdo é representada por um objeto que contém o link para a visualização do conteúdo correspondente (vide Figura 7.13).

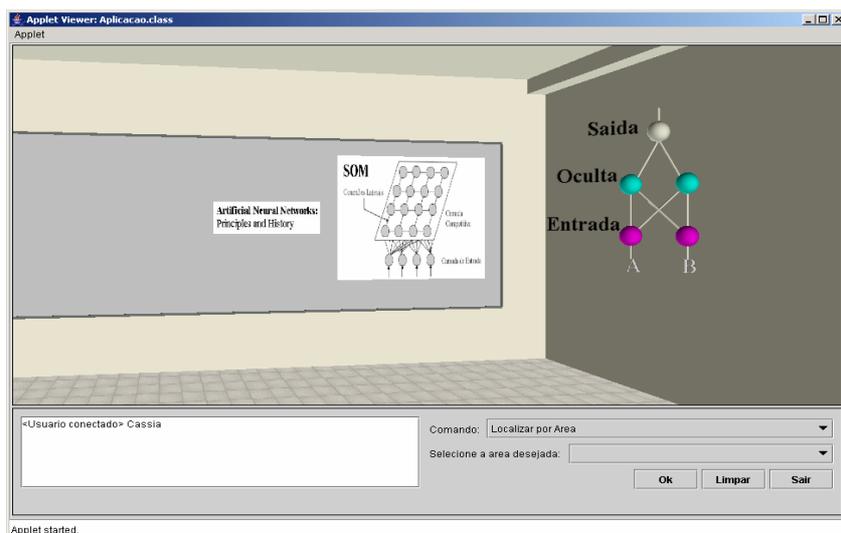


Figura 7.12 – Representação de conteúdos no ambiente.

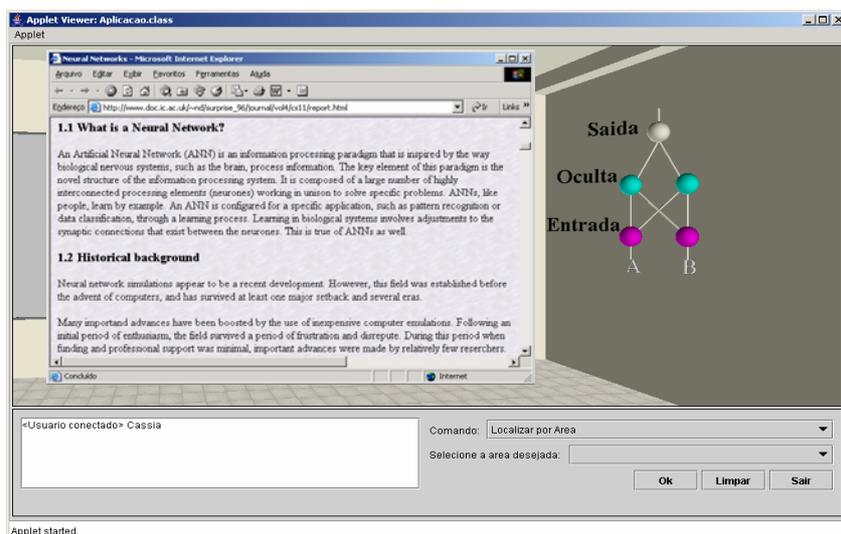


Figura 7.13 – Visualização de um conteúdo, após clique na descrição correspondente.

Por outro lado, quanto aos modelos dos conteúdos, o provedor atua na definição dos mesmos. O modelo do provedor é utilizado para indicar a área (Inteligência Artificial, por exemplo) a que pertence o conteúdo, sendo a categoria do conteúdo definida com base nas sub-áreas correspondentes. Conforme comentado na seção 6.2.2, os dados usados para a composição do modelo são definidos manualmente pelo provedor, através de um formulário. Entretanto, para conteúdos textuais, o processo de categorização automático pode ser invocado. A Figura 7.14 apresenta a interface de definição de um modelo de conteúdo, a partir da qual o usuário pode optar pela categorização manual ou automática do conteúdo. Além disso, é através desta interface que o conteúdo é inserido no ambiente, e ela é apresentada ao usuário a partir de uma

solicitação (para a inserção de um conteúdo no ambiente) do mesmo ao agente, conforme comentado na seção seguinte.

Figura 7.14 – Interface para definição do modelo de um conteúdo.

Quanto ao processo de categorização automático, foi seguida a metodologia apresentada na seção 3.4. Deste modo, foram coletados, a partir dos mecanismos de busca da Web, 30 artigos científicos para cada sub-área contemplada no ambiente. Após, foi realizado o pré-processamento dos exemplos e a geração dos scripts submetidos à ferramenta C4.5, responsável pela geração das Árvores de Decisão. Experimentos com as categorizações binária e múltipla foram realizados, utilizando-se 2/3 dos exemplos para o aprendizado do categorizador e 1/3 para a validação do mesmo, e aplicando-se o método *3-fold cross validation*²⁴ para estimar o erro de generalização na base de validação. Com o uso de um categorizador binário, um conteúdo pode ser associado a mais de uma sub-área (categoria), e com um categorizador múltiplo, um conteúdo é assinalado como pertencente a uma única sub-área. Deste modo, a cada área

²⁴ Método aplicado para estimar o erro de generalização que consiste na divisão da base de exemplos em k conjuntos de tamanho aproximadamente igual. O algoritmo de aprendizado é treinado k vezes e, em cada uma destas, um dos subconjuntos é utilizado para a validação.

pode ser associado um categorizador múltiplo, que indica a sub-área a que pertence um conteúdo, ou a cada sub-área pode ser associado um categorizador binário, que indica se determinado conteúdo pertence ou não a mesma. Neste último caso, um conteúdo poderá pertencer a mais de uma sub-área.

A Tabela 7.1 apresenta os resultados dos experimentos com os dois tipos de categorizadores. São indicados o erro de generalização (base de validação), o número de nodos da árvore gerada e as medidas de abrangência²⁵ e precisão²⁶, usadas para avaliar a performance da categorização. Os valores apresentados correspondem as médias dos erros de generalização, obtidas a partir do método *3-fold cross validation*. Para os categorizadores binários são mostradas as médias correspondentes aos categorizadores das sub-áreas.

As maiores taxas de erro foram obtidas com os categorizadores múltiplos e as áreas de CG e ES. Foi observado que as sub-áreas correspondentes a estas áreas, são as que mais possuem atributos comuns, dificultando a distinção das mesmas pelo categorizador. Por outro lado, as melhores taxas de classificação foram obtidas com o uso do categorizador múltiplo, na área de IA. Verificou-se que os atributos selecionados para a formação da árvore para esta área, foram os mais representativos de cada sub-área (*neural, artificial, algorithm, genetic, system, multi e agent*, por exemplo), permitindo uma exata distinção entre as mesmas e, conseqüentemente, boas taxas de categorização.

Tabela 7.1 – Comparativo entre os categorizadores binário e múltiplo.

Área	Categorização	Erro	Nodos	Abrangência	Precisão
CG	Múltipla	28,86	14,33	0,71	0,71
	Binária	19,25	8,85	0,78	0,80
ES	Múltipla	12,33	7,66	0,87	0,89
	Binária	9,25	6,78	0,89	0,91
IA	Múltipla	3,33	5,66	0,96	0,96
	Binária	5,18	5,44	0,94	0,94
RC	Múltipla	26,66	13,66	0,73	0,74
	Binária	13,71	9,22	0,86	0,87
Média	Múltipla	17,79	10,32	0,81	0,82
	Binária	11,84	7,57	0,86	0,88

²⁵ *Abrangência* é a fração de itens recuperados relevantes, em relação aos relevantes na base de dados.

²⁶ *Precisão* é a fração de itens recuperados relevantes, em relação ao total de recuperados.

No intuito de melhorar as taxas de erro obtidas com o categorizador múltiplo, especialmente em relação as áreas de CG e ES, foram realizados os experimentos com o categorizador binário. Pode-se observar, conforme a Tabela 7.1, que este categorizador apresentou, em geral, melhores resultados (menor erro e, conseqüentemente, melhor abrangência e precisão). Verificou-se que, conforme os atributos selecionados para a formação da maioria das árvores, houve uma melhor distinção entre os exemplos pertencentes e não pertencentes as categorias correspondentes. Exemplos comparativos das árvores geradas, para a área de RC, com os categorizadores múltiplo e binário podem ser visualizados nas Figuras 7.15 (a) e 7.15 (b). Conforme a árvore da Figura 7.15 (a), pode-se verificar que a palavra *network* foi utilizada na formação da árvore, mas no entanto, não correspondente a um atributo representativo das sub-áreas.

<pre> PROTOCOL <= 6 : SECUR > 10 : segurança (9.0) SECUR <= 10 : MANAG > 1 : gerenciamento (16.0) MANAG <= 1 : NETWORK <= 9 : segurança (4.0) NETWORK > 9 : gerenciamento (3.0/1.0) PROTOCOL > 6 : ATTACK > 1 : segurança (4.0) ATTACK <= 1 : MANAG <= 8 : protocolos (22.0/2.0) MANAG > 8 : gerenciamento (2.0/1.0) </pre>	<pre> PROTOCOL <= 6 : nao_protocolo (32.0) PROTOCOL > 6 : ATTACK > 1 : nao_protocolo (4.0) ATTACK <= 1 : MANAG <= 8 : protocolo (22.0/2.0) MANAG > 8 : nao_protocolo (2.0) </pre>
(a)	(b)

Figura 7.15 – (a) Árvore gerada a partir de um categorizador múltiplo (área RC); (b) Árvore gerada com um categorizador binário (área RC e sub-área Protocolos).

Portanto, considerando os resultados apresentados, o categorizador binário foi adotado no protótipo. Deste modo, a cada área, estão associados três categorizadores binários (um para cada sub-área). Assim, as árvores geradas para as sub-áreas, foram convertidas para regras do tipo IF – THEN, por uma aplicação em Java, e associadas ao módulo gerenciador de conteúdo. Na definição de um modelo de conteúdo textual, o provedor pode optar pela categorização automática e, neste caso, o conteúdo é pré-processado e as são regras avaliadas, podendo o mesmo ser atribuído a mais de uma sub-área. Para a adição de uma nova sub-área, são necessários apenas o treinamento e validação do categorizador correspondente a nova categoria, e a adição do mesmo ao módulo gerenciador, sem a necessidade de qualquer alteração nos categorizadores já existentes.

7.4 Agente Virtual Inteligente

Conforme comentado na seção 6.2.4, o agente virtual atua como assistente dos usuários na exploração do ambiente. O processo de comunicação entre o agente e o usuário está baseado em um conjunto de perguntas (solicitações do usuário) e respostas do agente. De forma a simplificar o processo, uma lista de opções, correspondentes as solicitações, é disponibilizada ao usuário. As respostas do agente são apresentadas na interface de comunicação e sintetizadas para voz, e podem envolver a movimentação do agente no ambiente. A interface de comunicação é formada por duas partes: um histórico do “diálogo” e uma seção com a listagem das opções de solicitação. A Tabela 7.2 apresenta as opções definidas para o protótipo e as respostas (do agente) correspondentes.

Tabela 7.2 – Opções de solicitações e respostas correspondentes.

Requerente	Agente
Localizar <área>	Indicar posição + movimentação até a sala
Localizar <sub-área>	Indicar posição + movimentação até a sub-sala
Localizar <palavras-chave>	Indicar conteúdos encontrados + posições (sub-salas)
Navegar <ambiente>	Apresentar áreas e sub-áreas + movimentação até cada sala
Navegar <área>	Apresentar sub-salas + movimentação até a sala e cada sub-sala
Provedor	Agente
Inserir conteúdo	Apresentar interface de inserção + movimentação até a sub-sala onde conteúdo foi inserido
Remover conteúdo	Apresentar interface de remoção
Atualizar conteúdo	Apresentar interface de atualização

A seguir, são apresentados exemplos da interação entre o usuário e o agente, conforme as possibilidades apresentadas na Tabela 7.2. Nas Figuras 7.16 e 7.17, são ilustradas interfaces onde o usuário solicita ao agente que ele localize a área de IA, e a localização, pelo agente, da área solicitada. Conforme comentado na seção 6.2.4, o agente mantém em sua base de conhecimento as informações sobre as localizações das áreas, sub-áreas e conteúdos, bem como um mapa topológico do ambiente. Deste modo, quando o usuário faz uma solicitação, o agente consulta a sua base de conhecimento e define as rotas que devem ser seguidas para apresentar ao usuário o que foi solicitado.

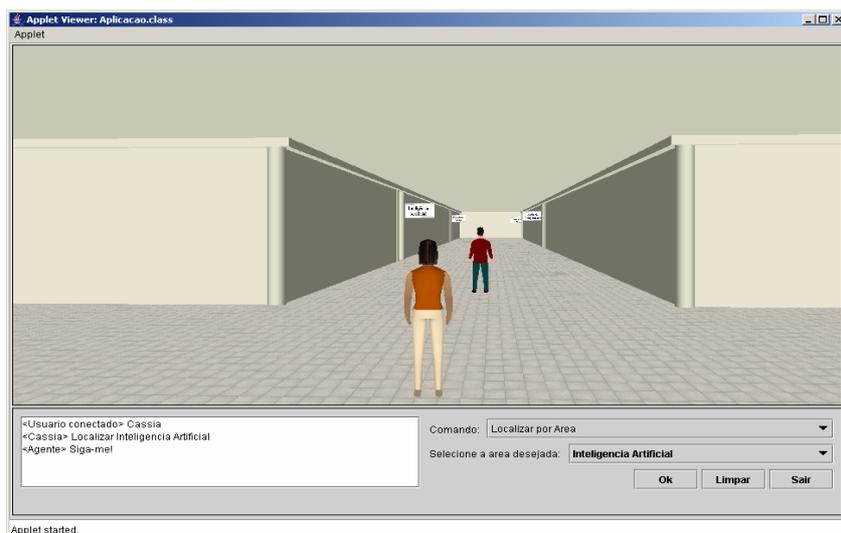


Figura 7.16 – Solicitação do usuário para a localização da área de IA.



Figura 7.17 – Localização, pelo agente, da área solicitada.

As Figuras 7.18 e 7.19 ilustram interfaces que contemplam a solicitação do usuário para a localização da sub-área de RNA, e a localização da mesma, pelo agente, respectivamente.



Figura 7.18 – Solicitação do usuário para a localização da sub-área de RNA.



Figura 7.19 – Localização, pelo agente, da sub-área solicitada.

Em relação a localização de conteúdos a partir de palavras-chave, o usuário informa as mesmas em um campo correspondente na interface de comunicação. O agente, então, consulta a sua base de conhecimento e recupera os conteúdos que satisfazem a solicitação do usuário. Neste processo, as palavras informadas são comparadas com as palavras-chave armazenadas nos modelos dos conteúdos. Os títulos dos conteúdos recuperados e as sub-salas onde os mesmos se encontram são listados na interface de comunicação. A Figura 7.20 apresenta um exemplo de interface onde o usuário informa as palavras-chave para a recuperação de conteúdos, e a Figura 7.21 ilustra um exemplo de interface onde são listados os títulos dos conteúdos recuperados.

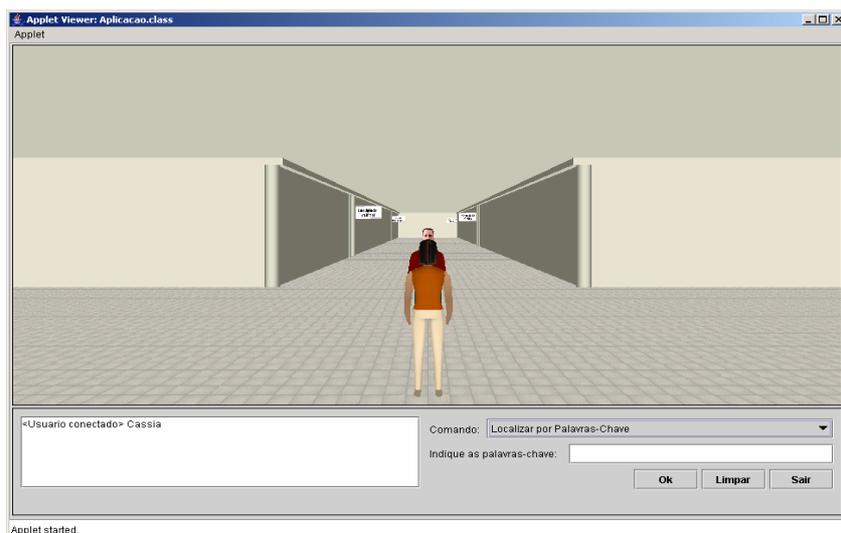


Figura 7.20 – Indicação das palavras-chave para a recuperação de conteúdos.

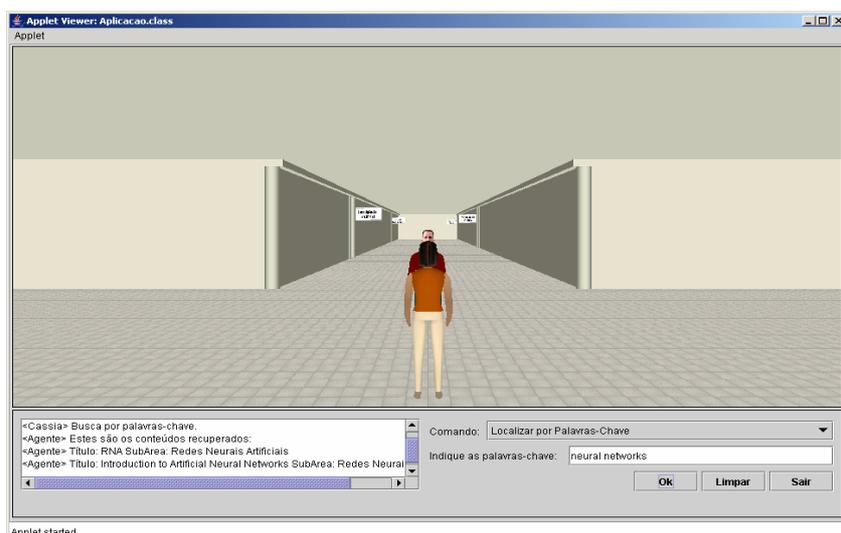


Figura 7.21– Listagem dos títulos e sub-salas correspondentes aos conteúdos recuperados.

Quando o usuário solicita auxílio para a navegação no ambiente, o agente apresenta as localizações das salas (áreas) e sub-salas (sub-áreas). Neste processo, o agente movimenta-se até cada sala, listando as descrições da área e sub-áreas correspondentes. A Figura 7.22 ilustra um exemplo de interface onde o agente apresenta o ambiente ao usuário, conforme uma solicitação de auxílio a navegação. Na navegação por uma área, o processo é similar: o agente desloca-se até a sala correspondente e apresenta as sub-salas, listando as descrições das sub-áreas na interface de comunicação.

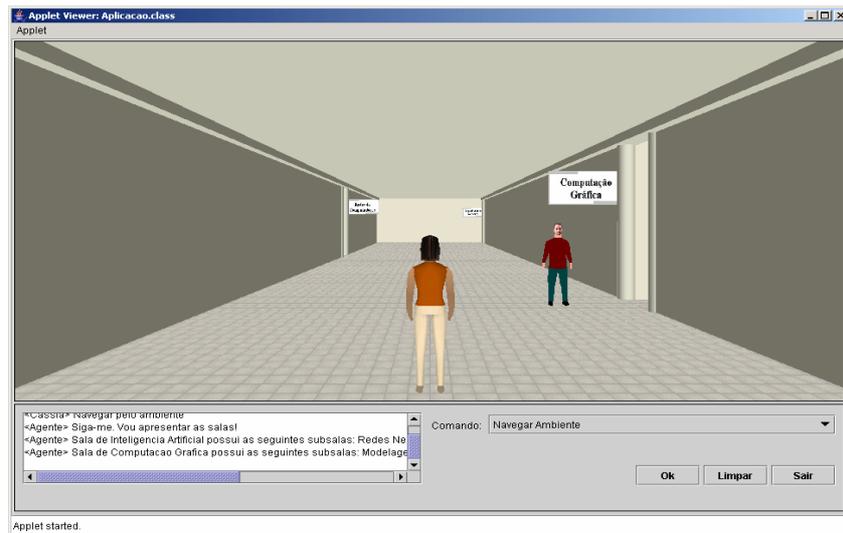


Figura 7.22 – Agente apresentando o ambiente ao usuário.

Para a inserção de conteúdos no ambiente, o agente apresenta ao provedor a interface correspondente. Nesta interface, o provedor especifica o modelo do usuário, podendo ser auxiliado pelo processo de categorização automático no caso dos conteúdos textuais. Após a especificação do modelo, o conteúdo é adicionado no ambiente, na sala correspondente a área principal indicada no modelo do provedor e na sub-sala que corresponde a categoria (sub-área) especificada no modelo do conteúdo. Em seguida, o agente conduz o provedor até uma das sub-salas onde o conteúdo foi adicionado. Considerando que um conteúdo pode estar associado a mais de uma categoria (sub-área), o mesmo pode ser adicionado em mais de uma sub-sala. O agente está apto a conduzir o provedor até a primeira das sub-salas (conforme a lista das sub-áreas do modelo do conteúdo) onde o conteúdo foi adicionado. Além disso, quanto ao tipo de conteúdo adicionado, os que envolvem a visualização externa ao ambiente (*.txt, *.doc, *.pdf, *.html, *.ppt e *.avi) são representados como objetos 3D com as descrições especificadas em seus modelos. O usuário, ao clicar no objeto, pode conferir o arquivo no visualizador correspondente. As figuras seguintes apresentam um exemplo do processo de inserção de um conteúdo no ambiente. A Figura 7.23 apresenta a interface onde o provedor solicita a inserção de um conteúdo no ambiente e o agente indica que o mesmo deve preencher o formulário que será apresentado.



Figura 7.23 – Interação para a inserção de um conteúdo no ambiente.

A Figura 7.24 apresenta o formulário para a especificação do modelo do conteúdo e inserção do conteúdo no ambiente. No exemplo, o provedor optou pela categorização automática do conteúdo, sendo obtidas a categoria e o conjunto de palavras-chave correspondentes. Pode ser observado que o provedor informou, para os campos *arquivo* e *tipo de mídia*, tipos de arquivos diferentes. O arquivo indicado no campo correspondente contém a descrição textual que será utilizada no processo de categorização, e o tipo de mídia indica a extensão do conteúdo que será inserido no ambiente. Neste caso, um conteúdo, com o nome indicado no campo *arquivo* e com a extensão indicada no campo *tipo de mídia*, é considerado na inserção. Deste modo, o arquivo *doc86.txt* é utilizado para fins de categorização, e o arquivo *doc86.pdf* corresponde ao conteúdo a ser inserido no ambiente. Considerando que se trata de um conteúdo *.pdf*, o campo *descrição* é utilizado na sua representação no ambiente. Este processo pode ser adotado para qualquer um dos tipos de mídia suportados no ambiente.

Figura 7.24 – Formulário para a especificação do modelo do conteúdo.

Após o preenchimento do formulário, o agente indica a sub-sala onde o conteúdo foi inserido e conduz o provedor até a mesma (Figuras 7.25 e 7.26). Por fim, o provedor pode visualizar o objeto que representa o conteúdo adicionado (Figura 7.27).



Figura 7.25 – Agente conduzindo usuário até a sub-sala onde o conteúdo foi inserido.

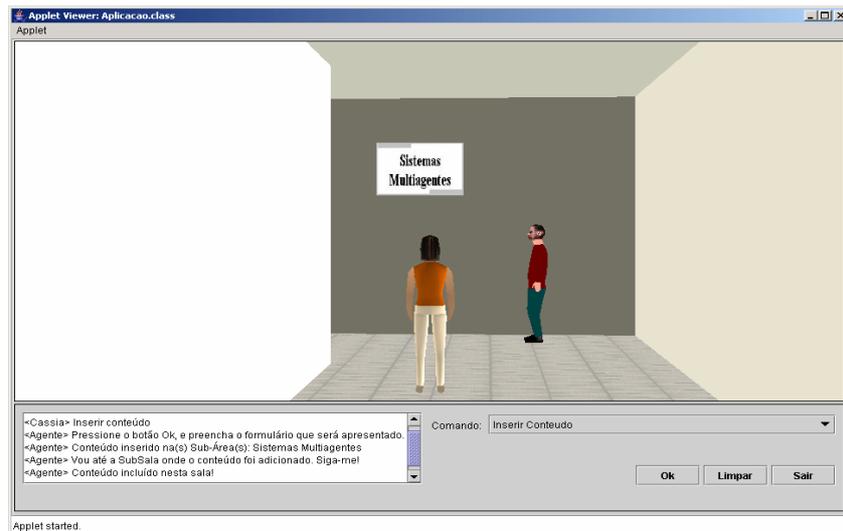


Figura 7.26 – Agente apresentando a sub-sala onde o conteúdo foi inserido.

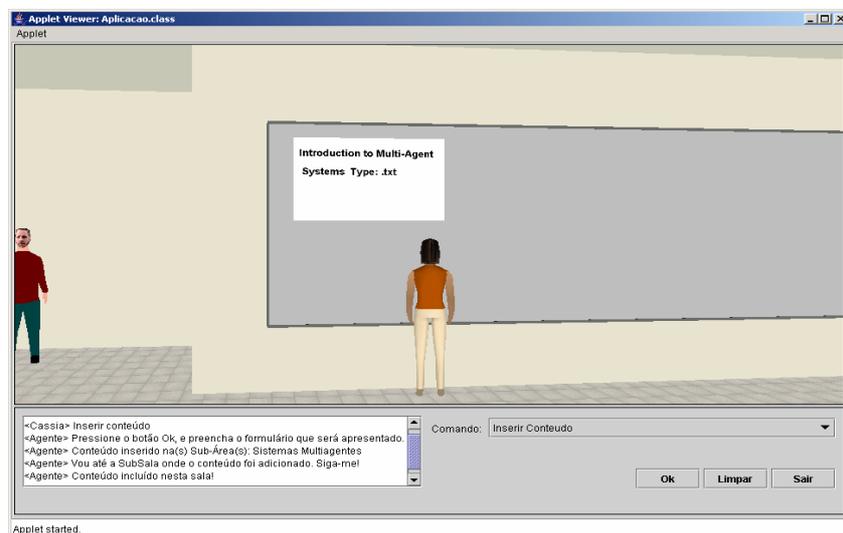


Figura 7.27 – Provedor visualizando o objeto que representa o conteúdo adicionado.

Por outro lado, de forma que o agente possa movimentar-se pelo ambiente, conforme apresentado nos exemplos acima, um mapa topológico do ambiente é mantido em sua base de conhecimento. Este mapa possui um conjunto de rotas que leva a cada sala e sub-sala do ambiente. As rotas são geradas automaticamente, conforme as distâncias entre as salas e sub-salas, previamente estabelecidas pelo *Gerador de ambientes*. Deste modo, a geração das rotas não exige qualquer esforço manual (independente do número de salas e sub-salas do ambiente). Cada rota corresponde a um conjunto de segmentos e posições-chave, conforme ilustrado na Figura 7.28. De acordo com a figura, a rota que leva à primeira sub-sala da sala de número 3, por exemplo, é formada pelos segmentos que compreendem como pontos principais, 1, 2 3 e 4. O

cálculo destes pontos é feito automaticamente para cada sala e sub-sala e, quando o agente desloca-se de uma sala para outra, ele computa a rota a ser seguida, a partir do ponto onde se encontra e os pontos que formam a rota até posição destino. Deste modo, de acordo com as informações sobre as localizações dos conteúdos (atualizadas a cada modificação no ambiente) e do mapa do ambiente, o agente está sempre apto a conduzir o usuário pelas novas localizações de salas, sub-salas ou conteúdos.

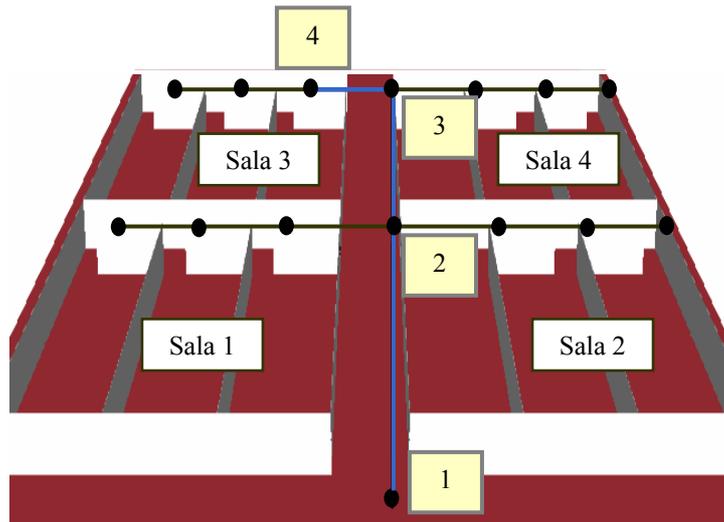


Figura 7.28 – Mapa topológico do ambiente.

Além disso, as respostas do agente, enviadas ao usuário através da interface textual de comunicação, são sintetizadas para voz. Para isto, um sintetizador texto-voz, foi desenvolvido, utilizando-se o padrão JSAPI²⁷ (*Java Speech API*).

Ainda, conforme comentado na seção 7.1, é disponibilizado ao usuário um mapa 2D do ambiente. Através deste mapa o usuário pode verificar a sua localização no ambiente (avatar que o representa), bem como a posição onde se encontra o agente. A Figura 7.29 apresenta interfaces do ambiente 3D e do mapa 2D correspondente, onde as setas apontam para as posições correntes do usuário (avatar) e do agente. O mapa 2D é disparado junto a interface correspondente ao ambiente 3D e pode ser desativado pelo usuário.

²⁷ <http://java.sun.com/products/java-media/speech/index.jsp>

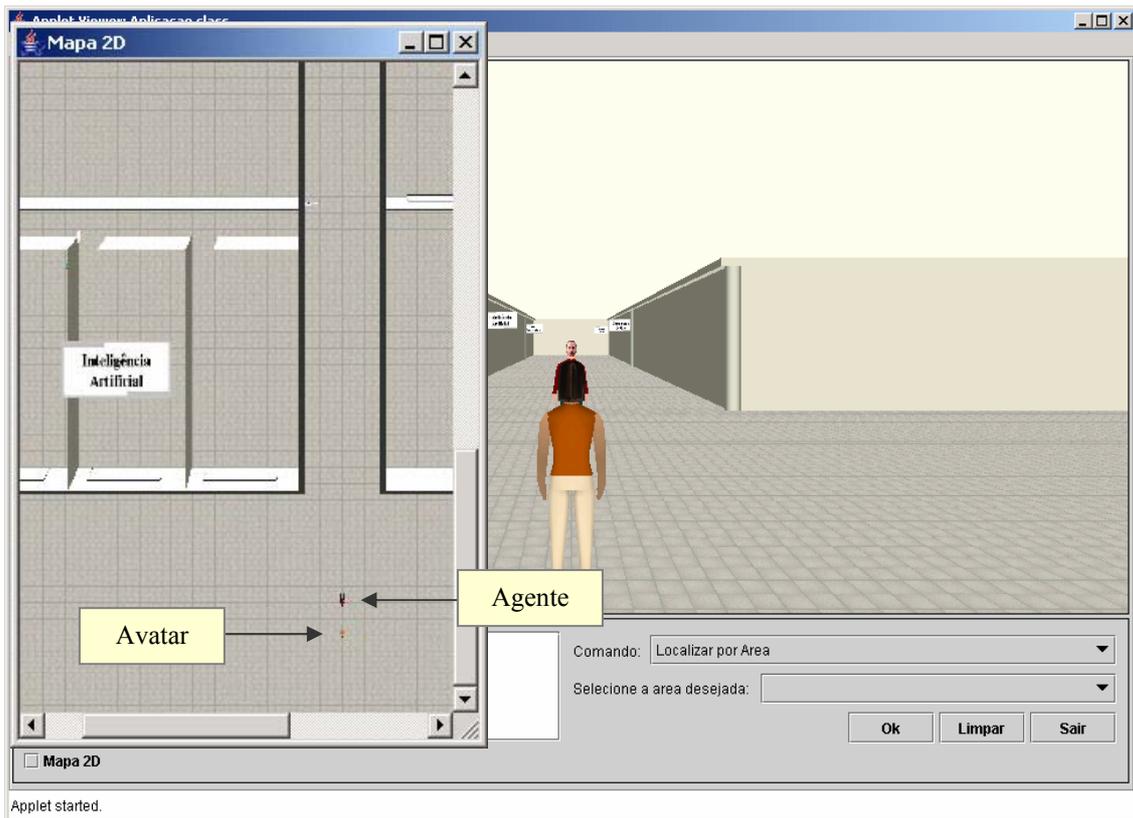
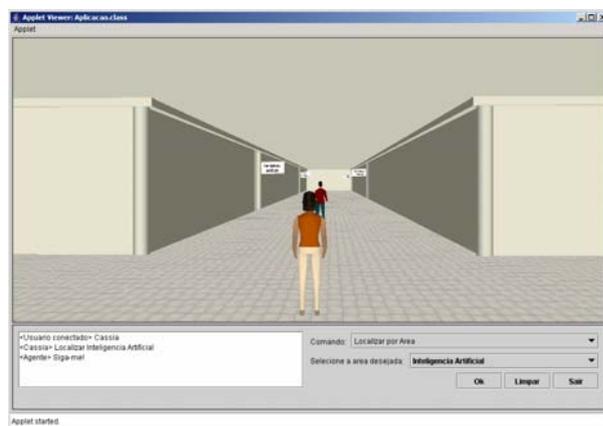
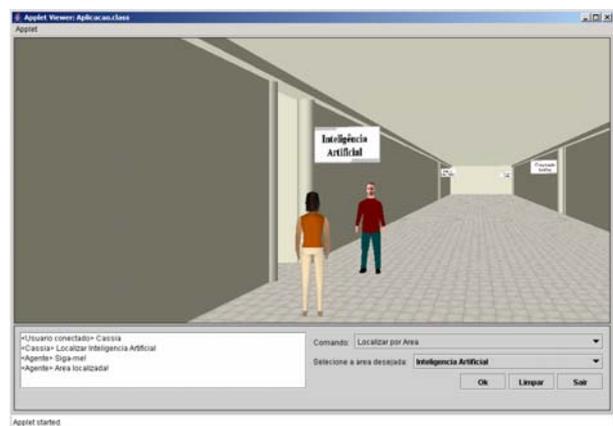


Figura 7.29 – Ambiente 3D e mapa 2D correspondente.

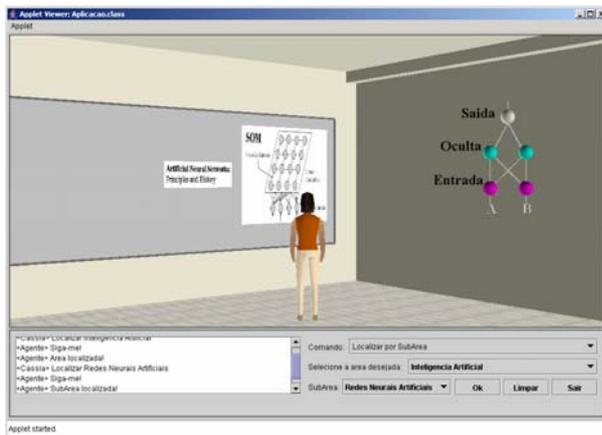
Para finalizar esta seção, é apresentado um exemplo completo de interação entre o usuário e o agente. As Figuras 7.30 (a), (b), (c) e (d) ilustram, respectivamente, uma solicitação do usuário para a localização de determinada área e a movimentação do agente; a localização de uma área pelo agente; o usuário visualizando conteúdos; e a visualização de detalhes de um conteúdo, após clique na descrição do conteúdo correspondente.



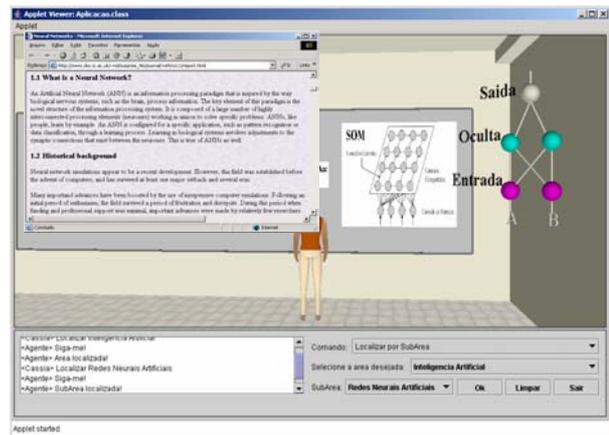
(a)



(b)



(c)



(d)

Figura 7.30 – (a) Solicitação do usuário ao agente; (b) localização de uma área pelo agente; (c) visualização de conteúdos da sub-área RNAs (IA); (d) visualização de detalhes de um conteúdo, após clique em uma descrição de conteúdo.

7.5 Extensão da Estrutura Inicial

De forma a verificar a robustez, especialmente do módulo *gerenciador de ambiente*, foram adicionadas novas áreas ao ambiente (Pesquisa Operacional – PO, Banco de Dados – BD, Computação Científica – CC, Informática na Educação – IE, Tolerância a Falhas –TF – e Processamento de Alto Desempenho – PAD). Às novas áreas foram associadas sub-áreas genéricas (“Sub-área 1”, “Sub-área 2”, “Sub-área 3”), sendo que os categorizadores automáticos destas sub-áreas ainda não foram disponibilizados (seria necessária apenas a coleta de textos e a geração automática dos novos categorizadores). A adição das novas áreas e sub-áreas implicou apenas na inserção dos dados correspondentes no banco de dados, sendo a estruturação do ambiente realizada de forma automática. A Figura 7.31 ilustra uma interface do ambiente com as novas áreas adicionadas, e a Figura 7.32 apresenta o mapa 2D correspondente.

- representação do usuário no ambiente (avatar) e mecanismo para a navegação pelo mesmo;
- aquisição e representação de um modelo inicial do usuário;
- mecanismo para a atualização do modelo inicial do usuário;
- aquisição e representação de um modelo de conteúdo, baseada no processo de categorização automático de conteúdos;
- ferramenta para auxílio à inserção de conteúdos no ambiente;
- agente inteligente como assistente dos usuários;
- navegação automatizada do agente pelo ambiente;
- mecanismo para comunicação entre os usuários e o agente;
- sintetização de texto-voz (agente);
- ferramenta para a construção de ambientes adaptados, conforme modelos de usuário e de conteúdo;
- suporte a diversos tipos de mídia no ambiente; e
- mapa 2D do ambiente.

Deste modo, os objetivos propostos para esta dissertação, conforme descrito no capítulo 1, foram alcançados. No capítulo seguinte, são apresentadas as considerações finais e as perspectivas de trabalhos futuros.

8 Conclusão

Os Ambientes Virtuais Inteligentes abrangem um campo de pesquisa que combina as tecnologias para a construção de ambientes 3D e Inteligência Artificial/Vida Artificial. Este campo envolve diferentes grupos de pesquisa concentrados em diversos aspectos gráficos do ambiente e da inteligência embutida neste. De um lado, os pesquisadores de Computação Gráfica estão interessados em tornar os seus ambientes mais dinâmicos e interessantes. Ambientes imutáveis e com animações pré-definidas, que não se alteram ao longo das interações com o usuário, podem limitar o interesse do mesmo. Por exemplo, cidades virtuais sem humanos virtuais podem fazer com que o usuário não tenha real senso de presença, e entidades com comportamentos inalteráveis podem tornar a interação rapidamente desinteressante. Por outro lado, os pesquisadores de IA e VA estão reconhecendo os AVs como uma opção interessante para testar suas técnicas (interações em tempo real em um ambiente visual podem ser mais interessantes do que as baseadas em interfaces textuais). Por exemplo, sistemas tutores podem explorar o uso de personagens tutores e interfaces inteligentes multi-modais, em lugar das interfaces bidimensionais, geralmente utilizadas.

Uma das principais contribuições deste trabalho é a arquitetura AdapTIVE, que contempla um ambiente virtual inteligente que tem a sua estrutura e apresentação adaptadas de acordo com modelos de usuário e de conteúdo. Um agente virtual inteligente atua como assistente dos usuários na navegação pelo ambiente e localização de conteúdos. O propósito da adaptação está em facilitar o acesso do usuário aos conteúdos, a partir da alteração da disposição espacial dos mesmos no ambiente, conforme os interesses do usuário. Cabe destacar, entretanto, que uma modificação brusca no ambiente pode causar a desorientação do usuário. Deste modo, as modificações propostas são sempre sugeridas ao usuário e realizadas apenas sob o seu consentimento. Além disso, são disponibilizados recursos que provêm auxílio à navegação e à localização de informações no ambiente, tais como o mapa 2D e o próprio agente inteligente, evitando que os usuários sintam-se “perdidos” no ambiente.

Deste modo, no ambiente AdapTIVE três componentes receberam especial atenção: modelos de usuários, modelos de conteúdos e agentes inteligentes.

Um modelo do usuário compreende o conhecimento que o sistema detém sobre o usuário, utilizado com o propósito de melhorar a interação. Grande parte das aplicações que provêm funcionalidades adaptativas está concentrada na aquisição e aplicação destes modelos em ambientes bidimensionais. O uso de modelos de usuários em ambientes tridimensionais é ainda pouco explorado, mas promissor. Deste modo, uma das contribuições desta dissertação está relacionada com a aquisição de modelos de usuários em um ambiente tridimensional e o uso destes na personalização do ambiente. Nestes ambientes, diferentemente de ambientes bidimensionais, onde assume-se que os conteúdos das páginas acessadas pelo usuário foram vistos por ele, é possível traçar melhor que objetos foram visualizados, verificando, por exemplo, se o usuário aproximou-se o suficiente de determinado objeto e se a câmera (que representa a direção da visão do usuário em relação ao espaço 3D) foi posicionada na direção do objeto. Assim, a interpretação da interação do usuário, em um ambiente tridimensional, pode ser mais intuitiva do que em ambientes bidimensionais.

A abordagem para a modelagem de usuários proposta nesta dissertação considerou os requisitos fundamentais associados ao processo, comentados no capítulo 2. Na aquisição dos dados para a composição do modelo foram utilizados os métodos explícitos, de forma que um conjunto suficiente de dados esteja disponível para a composição de um modelo inicial (aplicado nas primeiras interações), e implícito, permitindo a atualização do modelo, conforme os novos interesses dos usuários. Para a atualização do modelo, foram consideradas as incertezas relacionadas as hipóteses de novos interesses, a partir da utilização de regras e fatores de certeza.

Além disso, a definição de alguns parâmetros – número de sessões para a revisão do modelo, peso associado à evidência de acesso, persistência do acesso, e limiares para aumento e diminuição da $P(H)$ – permitiu o fino ajuste do processo de atualização dos modelos. A partir da janela de revisão, pôde-se configurar o intervalo de sessões a partir do qual o modelo é revisado. Em relação aos parâmetros peso e persistência da evidência de acesso, foi possível determinar o grau de relevância desta evidência na quantificação do interesse do usuário pela área correspondente, e a duração deste grau durante o curso da interação. Por fim, o uso de limiares para aumento e diminuição da

P(H) possibilitou a atualização dos valores iniciais correspondentes, conforme as evidências coletadas.

Assim, foi apresentada e implementada uma possível forma de interpretar a interação do usuário em um ambiente tridimensional, a partir da quantificação do grau de interesse do mesmo pelas áreas contempladas no ambiente, a qual está baseada na coleta de evidências (solicitação, navegação e acesso) e uso de regras e fatores de certeza.

Por outro lado, foi proposta a modelagem de conteúdos como uma tarefa associada a categorização automática de conteúdos, aplicada na organização dos mesmos no espaço tridimensional. A abordagem de categorização automática vem sendo adotada em contextos mais gerais e é ainda inexplorada em AVIs. Nesta dissertação, após um conjunto de experimentos iniciais, envolvendo Redes Neurais Artificiais e Árvores de Decisão, optou-se pelas ADs para a geração do categorizador. As ADs apresentaram taxas de erro satisfatórias e mostraram-se adequadas ao processo, pois são menos sensíveis as variações de parâmetros do que as RNAs, e o conhecimento gerado é de fácil compreensão, utilização e expansão, considerando que a árvore gerada pode ser facilmente convertida em regras do tipo IF-THEN e a adição de novas categorias implica apenas na adição de novas regras, de forma incremental.

Entretanto, cabe destacar que melhor desempenho, com menores taxas de erro, possivelmente poderia ser obtido com a aplicação de outros algoritmos de pré-processamento, envolvendo, por exemplo, o uso de métodos para a seleção de atributos baseados na frequência relativa, TF-IDF ou análise morfossintática. O Grupo de Inteligência Artificial (GIA) do Programa Interdisciplinar de Pós-Graduação em Computação Aplicada (PIPICA) tem realizado estudos de diferentes abordagens para o pré-processamento, categorização e clusterização de textos, objetivando verificar a que oferece melhor desempenho. Deste modo, os resultados obtidos pelo grupo poderiam ser aplicados na extensão do processo de categorização proposto nesta dissertação.

Além disso, foram utilizados categorizadores binários, a partir dos quais um conteúdo pode ser associado a mais de uma categoria (sub-área). Deste modo, a cada categoria está associado um categorizador binário, o qual indica se determinado conteúdo pertence ou não a mesma. O uso deste tipo de categorizador facilita a adição

de novas categorias, pois, neste caso, são necessários apenas o treinamento e a validação do categorizador correspondente a nova categoria, sem a necessidade de qualquer alteração nos categorizadores já existentes, o que não ocorre com os categorizadores múltiplos, onde os mesmos devem ser novamente treinados e validados.

Por fim, a abordagem de agentes vêm sendo adotada nas mais diversas áreas e aplicações. Atualmente, atenção tem sido atribuída ao uso de agentes inteligentes em ambientes virtuais. Nestes ambientes, o agente pode atuar como assistente do usuário na exploração do ambiente, sendo dada ênfase para certas propriedades dos agentes inteligentes, tais como percepção, habilidade social, reatividade, capacidade de raciocínio, adaptabilidade e representação gráfica. A partir da interação com o usuário (habilidade social), de sua percepção, capacidade de raciocínio e reatividade, o agente reconhece as solicitações do usuário, decide como agir e atua no ambiente (através da movimentação de sua representação gráfica). Com base em sua adaptabilidade, é capaz de atualizar o seu conhecimento a partir das modificações que ocorrem no ambiente. O uso de assistentes virtuais enriquece a interação com o ambiente virtual, torna o ambiente menos intimidador e mais natural ao usuário, e evita que os usuários sintam-se perdidos no ambiente.

O agente virtual proposto nesta dissertação atua como um agente de interface que assiste os usuários na localização de informações e navegação pelo ambiente. Ele possui as características citadas acima e atua de forma isolada no ambiente (não há interação com outros agentes de software). A característica de pró-atividade do agente não foi explorada em sua totalidade, o que envolveria a re-organização da arquitetura do agente para refletir, por exemplo, desejos e intenções. No ambiente, o agente está apto a verificar se o usuário está seguindo-o ou não, a partir de uma solicitação de auxílio à navegação pelo mesmo. No entanto, outros mecanismos de pró-atividade podem ser explorados: agente ir ao encontro do usuário, quando o mesmo solicita auxílio; indicação de que o ambiente está sem espaço para a adição de novos conteúdos, recomendando a remoção de alguns deles; detecção de que o usuário não visualizou um conteúdo adicionado; entre outros.

Desta forma, conforme comentado acima, o ambiente AdapTIVE combina, em um espaço tridimensional, o uso de modelos de usuários, a aplicação de modelos de conteúdos e a atuação de agentes inteligentes. Grande parte dos esforços em AVIs não

prevê a combinação destes componentes, sendo abordado, geralmente, apenas um subconjunto. Além disso, apesar das abordagens propostas (técnicas de aprendizado de máquina em categorização de textos, regras e fatores de certeza) serem adotadas em diversas aplicações, são ainda inexploradas em AVIs, sugerindo uma interessante aplicação das mesmas neste contexto.

Por fim, considerando que o ambiente integra três áreas que ainda apresentam muitas questões em aberto, o resultado prático foi um protótipo, desenvolvido para validar a arquitetura proposta. O protótipo consistiu em um ambiente de apoio a Educação a Distância (EaD), utilizado para a manter e organizar material didático. Não foram considerados mecanismos para avaliar o usuário que interage no ambiente, sendo dada ênfase às questões relacionadas à adaptabilidade do ambiente. Este tipo de ambiente apresentou-se ideal para a validação da arquitetura, pois envolve atualizações contínuas de conteúdos, uma diversidade de modelos de usuários e permite a exploração de sofisticadas representações dos conteúdos disponibilizados (usuário imerso em um espaço 3D e conteúdos organizados de maneira espacial).

No entanto, a arquitetura poderia ser estendida para suportar aplicações em diversas áreas, notadamente em aplicações de comércio eletrônico. Por exemplo, em uma loja virtual os produtos podem ser organizados por seção (categorias, tais como bazar, eletrônicos, outras) e dispostos no ambiente conforme os interesses dos usuários, e o agente pode atuar como um assistente de vendas.

Como trabalhos futuros, diversas possibilidades podem ser exploradas. Primeiro, a comunicação entre os usuários e o agente poderia ser estendida para suportar um diálogo do tipo *chatbot*. Segundo, a característica de pró-atividade do agente deveria ser melhor explorada, permitindo que o mesmo atuasse no ambiente com maior grau de autonomia. Terceiro, o planejamento do movimento do agente pelo ambiente poderia ser feito de forma automática, através do uso de um algoritmo de planejamento de trajetória, dispensando o uso do mapa topológico pré-definido. Quarto, a arquitetura proposta poderia ser adotada em uma aplicação de comércio eletrônico, explorando a criação de lojas virtuais adaptáveis, uso de assistentes de vendas e ferramentas para auxílio a disposição espacial das informações no ambiente. Quinto, a interação multi-usuário poderia ser suportada, onde estudos sobre a modelagem de grupos de usuários e das possibilidades de adaptação neste tipo de ambiente deveriam ser realizados. Sexto, o

uso de algoritmos alternativos de pré-processamento para a categorização de documentos poderia ser realizado, visando melhorar a exatidão da categorização apresentada nesta dissertação. Sétimo, o controle da inserção de conteúdos no ambiente poderia ser feito, de modo que pudessem ser detectadas possíveis duplicidades de conteúdos. Ao fim, os modelos de conteúdos poderiam ser representados através de ontologias.

Bibliografia Referenciada

- [Abbattista et al, 2002] Abbattista, F.; Degemmis, M; Fanizzi, N.; Licchelli, O. Lops, P.; Semeraro, G.; Zambetta, F. *Learning User Profile for Content-Bases Filtering in e-Commerce*. Workshop Apprendimento Automatico: Metodi e Applicazioni, Siena, 2002.
- [Adomavicius and Tuzhilin, 2001] Adomavicius, G. and Tuzhilin, A. *Using Data Mining Methods to Build Customer Profiles*. IEEE Computer, vol. 34, no. 2, pp. 74-82, 2001.
- [Agius and Angelides, 2001] Agius, H. and Angelides, M. *Modelling Content for Semantic-Level Querying of Multimedia*. Multimedia Tools and Applications, vol. 15, no. 1, pp. 5-37, 2001.
- [Anastassakis et al, 2001] Anastassakis, G.; Ritching, T.; Panayiotopoulos, T. *Multi-agent Systems as Intelligent Virtual Environments*. LNAI 2174, pp. 381-395, 2001.
- [Avradinis et al, 2000] Avradinis, N.; Vosinakis, S.; Panayiotopoulos, T. *Using Virtual Reality Techniques for the Simulation of Physics Experiments*. Proceedings of the 4th Systemics, Cybernetics and Informatics International Conference, Florida, USA, July, 2000.
- [Aylett and Cavazza, 2001] Aylett, R. and Cavazza, M. *Intelligent Virtual Environments – A state of the art report*. Eurographics 2001 Conference, Manchester, UK, 2001.
- [Aylett and Luck, 2000] Aylett, R. and Luck, M. *Applying Artificial Intelligence to Virtual Reality: Intelligent Virtual Environments*. Applied Artificial Intelligence, vol. 14, no. 1, pp. 3-32, 2000.
- [Balabanovic and Shoham, 1997] Balabanovic, M and Shoham Y. *Content-Based Collaborative Recommendation*. Communication of the ACM, vol. 40, no. 3, pp. 66-72, 1997.

- [Ballegooij and Eliëns, 2001] Ballegooij, A. and Eliëns, A. *Navigation by Query in Virtual Worlds*. Proceedings of the 6th International Conference on 3D Web Technology, Germany, February, 2001.
- [Basu et al, 1998] Basu, C.; Hirsh, H.; Cohen, W. *Recommendation as Classification: Using Social and Content-Based Information in Recommendation*. Proceedings of the 15th National Conference on Artificial Intelligence, Madison, Wisconsin, United States, July, 1998.
- [Bauer, 1996] Bauer, M. *Acquisition of User Preferences for Plan Recognition*. Proceedings of the 5th International Conference on User Modeling, Kailua-Kona, Hawaii, USA, 1996.
- [Bender and Osório, 2003] Bender, T. and Osório, F. *Reconhecimento e Recuperação de Imagens Utilizando Redes Neurais Artificiais do Tipo MLP*. Anais do 4º Encontro Nacional de Inteligência Artificial, Campinas, SP, 2003.
- [Bersot et al, 1998] Bersot, O.; Guedj, P.; Godéreaux, C.; Nugues, P. *A Conversational Agent to Help Navigation and Collaboration in Virtual Worlds*. *Virtual Reality*, vol. 3, no 1, pp. 71-82, 1998.
- [Billsus and Pazzani, 1999] Billsus, D. and Pazzani, M. *A Hybrid User Model for News Story Classification*. Proceedings of the 7th International Conference on User Modeling, Banff, Canada, pp. 99-108, 1999.
- [Bowman et al, 1998] Bowman, D.; Koller, D.; Hodges, L. *A Methodology for the Evaluation of Travel Techniques for Immersive Virtual Environments*. *Virtual Reality: Research, Development, and Applications*, vol 3, no. 2, pp.120-131, 1998.
- [Brenner et al, 1998] Brenner, W.; Zarnekow, R.; Witting, H. *Intelligent Software Agents: Foundations and Applications*. Springer-Verlag, 1998.
- [Brusilovsky, 2001] Brusilovsky, P. *Adaptive Hypermedia*. Contribution to the 10th Anniversary Issue of the Journal "User Modeling and User-Adapted Interaction", vol. 11, no. 1/2, 2001.

- [Carberry, 2001] Carberry, S. *Techniques for Plan Recognition*. User Modeling and User-Adapted Interaction, vol. 11, Kluwer Academic Publishers, pp. 31-48, 2001.
- [Castellano et al, 2001] Castellano, G.; Fanelli, A. M.; Roselli, T. *Mining Categories of Learns by a Competitive Neural Network*. Proceedings of the INNS – IEEE International Joint Conference on Neural Networks, Washington, July, 2001.
- [Chittaro and Ranon, 2000] Chittaro L., Ranon R., *Adding Adaptive Features to Virtual Reality Interfaces for E-Commerce*, Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-based Systems, Lecture Notes in Computer Science 1892, Springer-Verlag, Berlin, August, 2000.
- [Chittaro and Ranon, 2000a] Chittaro, L. and Ranon, R. *Virtual Reality stores for 1-to-1 e-commerce*. Proceedings of the Workshop on Designing Interactive Systems for 1-to-1 E-Commerce, The Hague, The Netherlands, 2000.
- [Chittaro and Ranon, 2002] Chittaro, L. and Ranon, R. *New Directions for the Design of Virtual Reality Interfaces to E-Commerce Sites*. Proceedings of the 5th International Conference on Advanced Visual Interfaces, ACM Press, New York, May, 2002.
- [Chittaro and Ranon, 2002a] Chittaro, L. and Ranon, R. *Dynamic Generation of Personalized VRML Content: a General Approach and its Application to 3D E-Commerce*. Proceedings of the 7th International Conference on 3D Web Technology, ACM Press, New York, February, 2002.
- [Chittaro et al, 2003] Chittaro, R.; Ranon, R.; Ieronutti, L. *Guiding Visitors of Web3D Worlds through Automatically Generated Tours*. Proceedings of the 8th International Conference on 3D Web Technology, ACM Press, New York, March 2003.
- [Chiu and Webb, 1999] Chiu, B. and Webb, G. *Dual-model: An Architecture for Utilizing Temporal Information in Student Modeling*. Proceedings of the 7th International Conference on Computers in Education, IOS Press, Amsterdam, Netherlands, pp. 111-118, 1999.
- [Correa, 1994] Correa, Filho. *A arquitetura de diálogos entre agente cognitivos distribuídos*. Rio de Janeiro: UFRJ, 1994 (Tese de doutorado).

- [Correa and Ludemir, 2002] Correa, R. and Ludemir, T. *Categorização Automática de Documentos: Estudo de Caso*. Proceeding of 16th Brazilian Symposium on Neural Networks, Porto de Galinhas, PE, Brazil, 2002.
- [Demazeau, 1995] Demazeau, Y. *From Interactions to Collective Behaviour in Agent-Based Systems*. Proceedings of the 1st. European Conference on Cognitive Science, Saint-Malo, France, 1995.
- [Duda and Hart, 1973] Duda, R. and Hart, P. *Pattern Classification and Scene Analysis*. John & Sons, New York, 1973.
- [Duarte et al, 2002] Duarte, E.; Braga, A.; Braga, J. 2002. *Agente Neural para Coleta e Classificação de Informações Disponíveis na Internet*. Proceeding of 16th Brazilian Symposium on Neural Networks, Porto de Galinhas, PE, Brazil, 2002.
- [Fischer, 2001] Fischer, G. *User Modeling in Human-Computer Interaction*. Contribution to the 10th Anniversary Issue of the Journal "User Modeling and User-Adapted Interaction", vol. 11, no. 1/2, 2001.
- [Franklin and Graesser, 1996] Franklin, S. and Graesser, A. *Is it an agent, or just a program? A taxonomy for autonomous agents*. Proceedings of the 3rd International Workshop on Agent Theories, Architectures and Languages, Springer-Verlag, Berlin, 1996.
- [Frery et al, 2002] Frery, A.; Kelner, J.; Moreira, J., Teichrieb, V. *Satisfaction through Empathy and Orientation in 3D Worlds*. CyberPsychology and Behavior, vol. 5, no. 5, pp. 451-459, 2002.
- [Frozza, 1997] Frozza, R. *Simula – Ambiente para Desenvolvimento de Sistemas Multiagentes Reativos*. Porto Alegre: UFRGS, 1997 (Dissertação de mestrado).
- [Garcia and Sichman, 2003] Garcia, A. and Sichman, J. *Agentes e sistemas multiagentes*. In: Rezende, S. *Sistemas Inteligentes: fundamentos e aplicações*. Barueri, SP: Manole. cap. 11, pp. 269-306, 2003.

- [Giraffa, 1997] Giraffa, L. *Seleção e Adoção de Estratégias de Ensino em Sistemas Tutores Inteligentes*. Porto Alegre: UFRGS, 1997 (Proposta de tese de doutorado).
- [Giarratano and Riley, 1998] Giarratano, J. and Riley, G. *Expert Systems – Principles and Programming*. 3 ed., PWS, Boston, 1998.
- [Goodwin, 1994] Goodwin, R. *Formalizing Properties of Agents*. Technical Report, School of Computer Science, Carnegie-Mellon University, Pittsburg, PA, September, 1994.
- [Grand and Cliff, 1998] Grand, S. and Cliff, D. *Creatures: Entertainment software agents with artificial life*. *Autonomous Agents and Multi-Agent Systems*, vol. 1, no. 1, pp. 39-57, 1998.
- [Gratch et al, 2002] Gratch, J.; Rickel, J.; Andre, E.; Badler, N.; Cassell, J.; Petajan, E. *Creating Interactive Virtual Humans: Some Assembly Required*. *IEEE Intelligent Systems*, vol. 17, no. 4 (Special issue on AI in Interactive Entertainment), 2002.
- [Hayes-Roth, 1995] Hayes-Roth, B. *An architecture for adaptive intelligent systems*. *Artificial Intelligence (Special Issue on Agents and Interactivity)*, vol. 72, pp. 329-365, 1995.
- [Haykin, 2001] Haykin, S. *Redes Neurais: Princípios e Prática*. 2 ed. Bookman, 2001.
- [Horvitz et al, 1998] Horvitz, E.; Breese, J.; Heckerman, D.; Hovel, D.; Rommelse, K. *The Lumière Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users*. *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, Madison, WI: Morgan Kaufmann Publishers, 1998.
- [Jennings and Wooldridge, 1996] Jennings, N. and Wooldridge, M. *Software Agents*. *IEE Review*, January, 1996.
- [Joachims et al, 1997] Joachims, T.; Freitag, D.; Mitchell T. *WebWatcher: A Tour Guide for the World Wide Web*. *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, Nagoya, Japan, 1997.

- [Johansson, 2002] Johansson, P. *User Modeling in Dialog Systems*. St. Anna Report: SAR, 2002.
- [Kass and Finin, 1988] Kass, R and Finin, T. *Modeling the User in Natural Language Systems*. Computational Linguistics, vol. 14, no. 3, pp. 5-22, 1988. In: (Johansson 2002).
- [Kobsa, 1995] Kobsa, A. *Supporting User Interfaces for All Through User Modeling*. Proceedings of the HCI International, Yokohama, Japan, pp. 155-157, 1995.
- [Kobsa, 2001] Kobsa, A. *Generic User Modeling Systems*. Contribution to the 10th Anniversary Issue of the Journal "User Modeling and User-Adapted Interaction", vol. 11, no. 1/2, 2001.
- [Levy and Weld, 2000] Levy, A. and Weld, D. *Intelligent Internet Systems*. Artificial Intelligence, vol. 118, no. 1-2, pp. 1-14, 2000.
- [Lewis, 1991] Lewis, D. *Evaluating Text Categorization*. Proceedings of the Speech and Natural Language Workshop, pp. 312-318, Morgan-Kaufmann, February, 1991.
- [Lieberman, 1995] Lieberman, H. *Letizia: An Agent That Assist Web Browsing*. International Joint Conference on Artificial Intelligence, Montreal, pp. 924-929, 1995.
- [Maes, 1994] Maes, P. *Modeling Adaptive Autonomous Agents*. Artificial Life Journal, 1994.
- [Melcop et al, 2002] Melcop, T.; Costa, I.; Barros, F.; Ramalho, G. *Uma Ferramenta para Recuperação e Categorização de Páginas Web para Domínios Específicos*. 1º Workshop de Ontologias para a Construção de Metodologias de Buscas na Web - XIII Simpósio Brasileiro de Informática na Educação, São Leopoldo, RS, 2002.
- [Middleton et al, 2002] Middleton S.; Alani, H.; Shadbolt, N.; De Roure, C. *Exploiting Synergy between Ontologies and Recommender Systems*. Proceedings of the Semantic Web Whorkshop, Hawaii, USA, 2002.

- [Milde, 2000] Milde, J. *The instructable Agent Lokutor*. Working Notes – Autonomous Agents 2000, Workshop on Communicative Agents in Intelligent Virtual Environments, Barcelona, Spain, 2000.
- [Min et al, 2003] Min, P; Halderman, J.; Kazhdan, M.; Funkhouser, T. *Early Experiences with a 3D Model Search Engine*. Proceedings of the 8th International Conference on 3D Web Technology, ACM Press, New York, March, 2003.
- [Mitchell, 1997] Mitchell, T. *Instance-Based Learning*. Chapter 8 of Machine Learning, McGraw-Hill, 1997.
- [Musse, 2000] Musse, S. *Human Crowd Modelling with Various Levels of Behaviour Control*. Lausanne: EPFL, 2000. 164 p. (Tese de doutorado).
- [Ng et al, 1997] Ng, H.; Goh, W.; Low, K. *Feature Selection, Perceptron Learning and Usability Case Study for Text Categorization*. Proceedings of the 20th ACM International Conference on Research Development in Information Retrieval, Philadelphia, PA, USA, pp. 67-73, 1997.
- [Nijholt and Hulstijn, 2000] Nijholt, A. and Hulstijn, J. *Multimodal Interactions with Agents in Virtual Worlds*. In: Kasabov, N. (ed.): Future Directions for Intelligent Information Systems and Information Science, Physica-Verlag: Studies in Fuzziness and Soft Computing, 2000.
- [Nikolopoulos, 1997] Nikolopoulos, C. *Expert Systems – Introduction to First and Second Generation and Hybrid Knowledge Based Systems*. Eds.: Marcel Dekker, New York, 1997.
- [Noll et al, 1999] Noll, S.; Paul, C.; Peters, R.; Schiffner, N. *Autonomous Agents in Collaborative Virtual Environments*. IEEE 8th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, June, Palo Alto, California, 1999.
- [Nunez and Blake, 2003] Nunez, D. and Blake, E. *A direct comparison of presence levels in text-based and graphics-based virtual environments*. Proceedings of the 2nd International Conference on Computer Graphics, Virtual Reality, Visualization and Interaction in Africa, Cape Town, South Africa, 2003.

- [Nwana, 1996] Nwana, H. *Software Agents: An Overview*. In: Knowledge Engineering Review, vol. 11, no. 3, pp. 205-144, 1996.
- [Oliveira and Castro, 2000] Oliveira, C. and Castro, P. *Categorização Múltipla com Árvores de Decisão e Regras*. Instituto Militar de Engenharia, Departamento de Engenharia de Sistemas, 2000 (Relatório técnico).
- [Osorio and Amy, 1999] Osorio, F. and Amy, B. *INSS: A hybrid system for constructive machine learning*. Neurocomputing, vol. 28, pp. 191-205, 1999.
- [Orwant, 1995] Orwant, J. *Heterogeneous Learning in the Doppelgänger User Modeling System*. User Modeling and User-Adapted Interaction, vol.4, no. 2, pp. 107-130, 1995.
- [Paliouras et al, 1999] Paliouras, G.; Papatheodorou, C.; Karkaletsis, V.; Tzitziras, P.; Spyropoulos, C. *Learning Communities of the ACAI'99 Web-site Visitors*. Proceedings of the Workshop on Machine Learning in User Modeling, Advanced Course in Artificial Intelligence, Chania, Grece, 1999.
- [Paliouras et al, 1999a] Paliouras, G; Karkaletsis, V; Papatheodorou, C.; Spyropoulos, C. *Exploiting Learning Techniques for the Acquisition of User Stereotypes and Communities*. Proceedings of the 7th International Conference on User Modeling, Canada, June, 1999.
- [Panayiotopoulos et al, 1999] Panayiotopoulos, T.; Zacharis, N.; Vosinakis, S. *Intelligent Guidance in a Virtual University*. Advances in Intelligent Systems – Concepts, Tools and Applications, pp. 33-42, Kluwer Academic Press, 1999.
- [Papatheodorou, 2001] Papatheodorou, C. *Machine Learning in User Modeling*. Machine Learning and Applications. Lecture Notes in Artificial Intelligence. Springer Verlag, 2001.
- [Pazzani and Billsus, 1997] Pazzani, M. and Billsus, D. *Learning and Revising User Profiles: The identification on Interesting Web Sites*. Machine Learning, vol. 27, no. 3, pp. 313-331, 1997.

- [Pree and Koskimies, 1999] Pree, W. and Koskimies, K. *Framelets-Small Is Beautiful, A Chapter in Building Application Frameworks: Object Oriented Foundations of Framework Design*. Eds: M.E. Fayad, D.C. Schmidt, R.E. Johnson, Wiley & Sons, 1999.
- [Perkowitz and Etzioni, 1998] Perkowitz, M. and Etzioni, O. *Adaptive Web Sites: Automatically synthesizing Web pages*. Proceedings of the 15th National Conference in Artificial Intelligence, Wisconsin, 1998.
- [Perkowitz and Etzioni, 1999] Perkowitz, M. and Etzioni, O. *Adaptive Web Sites: Conceptual Cluster Mining*. Proceedings of the 16th International Joint Conference in Artificial Intelligence, Stockholm, Sweden, 1999.
- [Quinlan, 1986] Quinlan, R. *Induction of decision trees*. Machine Learning, vol. 1, pp. 81 – 106, 1986.
- [Quinlan, 1993] Quinlan, R. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, Sao Mateo, California, 1993.
- [Rao and Georgeff, 1995] Rao, A. and Georgeff, M. *BDI Agents: From Theory to Practice*. Proceedings of the 1st International Conference on Multi-Agents Systems, San Francisco, CA, June, 1995.
- [Reilly and Bates, 1992] Reilly, S. and Bates, J. *Building Emocional Agents*. Technical Report, School of Computer Science, Carnegie-Mellon University, Pittsburg, PA, May, 1992.
- [Reynolds, 1987] Reynolds, C. *Flocks Herds and Schools: A Distributed behavioural model*. Proceedings of the SIGGRAPH 87, Computer Graphics, vol. 21, no. 4, pp. 25-34, 1987.
- [Rich, 1983] Rich, E. *Users are Individuals: Individualizing User Models*. International Journal of Man-Machine Studies, vol. 18, pp. 199-214, 1983.
- [Rich, 1989] Rich, E. *Stereotypes and User Modeling*. User Modeling in Dialog Systems, Berlin, Heidelberg: Springer, pp. 35-21, 1989.

- [Rickel and Johnson, 1997] Rickel, J. and Johnson, W. *Integrating Pedagogical Capabilities in a Virtual Environment Agent*. Proceedings of the 1st International Conference on Autonomous Agents, February, ACM Press, 1997.
- [Rickel and Johnson, 2000] Rickel, J and Johnson, W. *Task-Oriented Collaboration with Embodied Agents in Virtual Worlds*. Embodied Conversational Agents, MIT Press, Boston, 2000.
- [Rickel et al, 2002] Rickel, J.; Marsella, S.; Gratch, J.; Hill, R.; Traum, D.; Swartout W. *Toward a New Generation of Virtual Humans for Interactive Experiences*. IEEE Intelligent Systems, vol. 17, no. 4 (Special issue on AI in Interactive Entertainment), 2002.
- [Rijsbergen, 1979] Rijsbergen, C. *Information Retrieval*. 2ed. London: Butterworths, 147p., 1979.
- [Rizzi et al, 2000] Rizzi, C.; Wives, L.; Oliveira, J., Engel, P. *Fazendo uso da Categorização de Textos em Atividades Empresariais*. International Symposium on Knowledge Management/Document Management, Curitiba, pp. 251-268, 2000.
- [Rizzo et al, 2002] Rizzo, A.; Bowely, T.; Buckwalter, G.; Schultheis, M.; Matheis, R.; Shahabi, C.; Neumann, U.; Kim, L.; Sharifzadeh, M. *Virtual Environments for the Assessment of Attention and Memory Processes: The Virtual Classroom and Office*. Proceedings of the International Conference on Disability, Virtual Reality and Associated Technology, Vesaprem, Hungary, September, 2002.
- [Russell and Norvig, 1995] Russell, S. and Norvig, P. *Artificial Intelligence: A Modern Approach*. Prentice Hall Series in Artificial Intelligence, New Jersey, 1995.
- [Salton and MacGill, 1983] Salton, G. and MacGill, M. *Introduction to Modern Information Retrieval*. New York: McGraw-Hill, 448 p., 1983.
- [Santos et al, 2002] Santos, C.; Frozza, R.; Paschoal, L.; Dahmer, A. *Dóris - Pedagogical Agent for Intelligent Tutoring System*. Proceedings of the 6th International Conference on Intelligent Tutoring System, Lecture Notes in Computer Science, Biarritz, France, 2002.

- [Schein et al, 2002] Schein, A.; Popescul, A., Ungar, L.; Pennock, D. *Methods and Metrics for Cold-Start Recommendations*. Proceedings of the 25th Annual International ACM Conference on Research and Development in Information Retrieval, pp. 253-160, Tampere, Finland, August, 2002.
- [Schwab and Kobsa, 2002] Schwab, I. and Kobsa, A. *Adaptivity through Unobstrusive Learning*. KI, vol. 16, no. 3 (Special Issue on Adaptivity and User Modeling), pp. 5-9, 2002.
- [Sebastiani, 2002] Sebastiani, F. *Machine learning in automated text categorization*. ACM Computing Surveys, vol. 34, no. 1, pp.1-47, 2002.
- [Self, 1999] Self, J. *The defining characteristics of intelligent tutoring systems research: ITSs care, precisely*. International Journal of Artificial Intelligence in Education, vol. 10, pp. 350-364, 1999.
- [Silva and Vieira, 2003] Silva, C. and Vieira, R. *Avaliação da Seleção de Atributos para Clusterização de Documentos*. Anais do Encontro Nacional de Inteligência Artificial, Campinas, SP, 2003.
- [Silva et al, 2003] Silva, C.; Perez, C.; Osório, F.;Vieira, R.; Goulart, R. *Informações sintáticas na Mineração de Textos*. 1º Workshop em Tecnologia da Informação e da Linguagem Humana, São Carlos, SP, 2003.
- [Sycara et al, 1996] Sycara, K.; Decker, K.; Pannu, A.; Williamson, M.; Zeng, D. *Distributed Intelligent Agents*. IEEE Expert, 1996.
- [Tecuci 1998] Tecuci, G. *Building Intelligent Agents: An Multistrategy Learning Theory, Methodology, Tools and Case Studies*. Academic Press, 320 p., 1998.
- [Teichrieb, 1999] Teichrieb, V. *Avatares como Guias Interativos para Auxílio na Navegação em Ambientes Virtuais Tridimensionais*. Recife: UFPE, 1999 (Dissertação de mestrado).
- [Terzopoulos et al, 1994] Terzopoulos, D; Tu, X.; Grzeszczuk, R. *Artificial fishes with autonomous locomotion, perception, behavior and learning, in a physical world*. In

- Maes, P. and Brooks, R., editors, Proceedings of the Artificial Life IV Workshop, MIT Press, pp. 17-27, 1994.
- [Theodoridas and Koutroumbas, 1998] Theodoridas, S. and Koutroumbas, K. *Pattern Recognition*. Academic Press, pp. 351-495, 1998.
- [Viccari and Giraffa, 1996] Viccari, R. and Giraffa, L. *Sistemas Tutores Inteligentes: abordagem tradicional x abordagem de agentes*. Proceedings of the 13th Brazilian Symposium on Artificial Intelligence, Curitiba, PR, Brazil, Springer Verlag, 1996.
- [Wasfi, 1999] Wasfi, A. *Collecting User Access Pattern for Building User Profiles and Collaborative Filtering*. Proceedings of the International Conference on Intelligent User Interface, Los Angeles, USA, pp. 57-64, 1999.
- [Webb et al, 2001] Webb, G.; Pazzani, M.; Billsus, D. *Machine Learning for user modeling*. User Modeling and User-Adapted Interaction, vol. 11, pp.19-29, 2001.
- [Wiener et al, 1995] Wiener, E.; Pedersen, J.; Weigend, A. *A Neural Network Approach to Topic Spotting*. Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval, pp. 22-34, 1995.
- [Wooldridge and Jennings, 1995] Wooldridge, M. and Jennings, N. *Intelligent Agents: Theory and Practice*. In: Knowledge Engineering Review, vol. 10, no. 2, 1995.
- [Yang and Pedersen, 1997] Yang, Y. and Pederson, J. *A Comparative Study on Feature Selection in Text Categorization*. Proceedings of the 14th International Conference on Machine Learning, pp. 412-420, Morgan Kaufmann Publishers, San Francisco, US, 1997.
- [Yang and Liu, 1999] Yang, Y. and Liu, X. *An evaluation of statistical approaches to text categorization*. Journal of Information Retrieval, vol.1, no.1/2, pp. 67-88, 1999.

Bibliografia Consultada

- [Jennings and Wooldridge, 1998] Jennings, N. and Wooldridge, M. *Applications of Intelligent Agents*. In: Agent Technology: Foundations, Applications and Markets, pp. 3-28, 1998.
- [Langley, 1999] Langley, P. *User Modeling in Adaptive Interfaces*. 7th International Conference on User Modeling, Banff, Canada, pp. 357-370, 1999.
- [Middleton, 2001] Middleton, S. *Interface agents: A review of the field*. Technical Report (AI Group), University of Southampton, 2001.
- [Mobasher et al, 2000] Mobasher, B.; Cooley, R.; Srivastava, J. *Automatic Personalization Based on Web Usage Mining*. Communications of the ACM, vol. 43, no. 8, pp. 142-151, 2000.
- [Mooney and Roy, 2000] Mooney R. and Roy, L. *Content-Based Book Recommending Using Learning for Text Categorization*. Proceedings of the 5th ACM Conference on Digital Libraries, San Antonio, USA, 2000.
- [Rashid et al, 2002] Rashid A.; Albert, I.; Cosley, D.; Lam, S.; MecNee, S.; Kostan, J.; Riedl, J. *Getting to Know You: Learning New User Preferences in Recommender Systems*. Proceedings of the International Conference of Intelligent User Interfaces, San Francisco, USA, 2002.
- [Tavares et al, 2001] Tavares, T.; Lemos, G.; Araújo, A.; Burlamaqui, A.; Jacinto, K.; Lucena, M. *Projeto e Implementação do ICSPACE – Um espaço cultural na Internet*. Proceedings of the 4th SBC Symposium on Virtual Reality, Florianópolis, SC, Brazil, October, 2001.