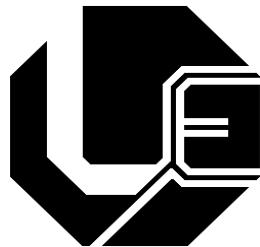


UNIVERSIDADE FEDERAL DE UBERLÂNDIA

FACULDADE DE ENGENHARIA ELÉTRICA

PROGRAMA DE PÓS-GRADUAÇÃO EM

ENGENHARIA ELÉTRICA



**MODELO HÍBRIDO DE SISTEMA TUTOR INTELIGENTE
UTILIZANDO CONHECIMENTO DO ESPECIALISTA E MAPAS DE
KOHONEN COM TREINAMENTO AUTOMATIZADO**

Sirlon Diniz de Carvalho

Orientadora: Profa. Dra. Edna Lúcia Flôres

Uberlândia
2012

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

SIRLON DINIZ DE CARVALHO

**MODELO HÍBRIDO DE SISTEMA TUTOR
INTELIGENTE UTILIZANDO CONHECIMENTO DO
ESPECIALISTA E MAPAS DE KOHONEN COM
TREINAMENTO AUTOMATIZADO**

Tese apresentada ao Programa de Pós-Graduação da Faculdade de Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial para a obtenção do título de Doutor em Engenharia Elétrica.

Área de Concentração: Engenharia Elétrica

Linha de Pesquisa: Processamento de Sinais

Orientadora: Profa. Dra. Edna Lúcia Flôres

Uberlândia
2012

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

SIRLON DINIZ DE CARVALHO

**MODELO HÍBRIDO DE SISTEMA TUTOR
INTELIGENTE UTILIZANDO CONHECIMENTO DO
ESPECIALISTA E MAPAS DE KOHONEN COM
TREINAMENTO AUTOMATIZADO**

Banca de Defesa:

Profa. Edna Lúcia Flôres, Dra. (UFU)
Presidente da Banca

Eliane Elias Ferreira dos Santos, Dra. (UFU)

Alzino Mendonça Furtado, Dr. (IFG)

Ricardo Antônio Gonçalves Teixeira, Dr. (UFG)

Francisco Ramos de Melo , Dr. (UEG)

Para minha família e amigos que, incondicionalmente, sempre estiveram ao meu lado.

Para minha esposa e filhos, por sempre terem sido fonte de inspiração.

AGRADECIMENTOS

A minha orientadora e amiga Edna Lúcia Flôres, pelos ensinamentos, companheirismo e por ter me desafiado sempre que precisei.

Ao meu amigo Francisco Ramos de Melo por ter sabido ser mestre, amigo e parceiro nos momentos em que precisei.

Ao meu amigo Luiz Fernando Batista Loja pelo apoio no desenvolvimento e orientações sobre a tecnologia adotada.

A todos os meus professores, de hoje e de ontem, que contribuíram para o meu crescimento.

Aos meus alunos por me propiciarem o desafio infindável da busca por conhecimentos.

RESUMO

A educação contemporânea tem muitos desafios, dentre eles o de adequar a utilização de novas tecnologias aos paradigmas clássicos de ensino. Com a educação a distância não tem sido diferente. Nesse contexto, este trabalho apresenta o processo de desenvolvimento de um sistema tutor inteligente híbrido com decisões baseadas no conhecimento do professor, subsidiadas pelas decisões das redes neurais *Self Organizing Maps* (SOM) ou Mapas de Kohonen. O sistema proposto tem a sua estratégia de ensino inicial estabelecida pelo professor especialista e à medida em que o sistema é utilizado a estratégia pedagógica é refinada pelas redes neurais, que utilizam padrões extraídos dos estudantes que usaram o tutor. O modelo propõe a utilização de uma estrutura de rede neural com treinamento automatizado que é capaz de treinar diferentes redes e definir a que apresente o resultado mais coerente com conjunto de padrões, dispensando a intervenção de um especialista na avaliação do desempenho das redes. O sistema tem característica adaptativa e reativa ao aprendiz, capaz de oferecer ao estudante um ensino personalizado e dinâmico. O sistema foi desenvolvido para ambiente web com o objetivo de obter vantagens que essa tecnologia oferece. Neste trabalho, além do desenvolvimento do modelo proposto também foram realizadas coletas de dados com alunos dos anos iniciais do ensino técnico integrado do Instituto Federal de Goiás – Campus Luziânia para avaliar a aplicabilidade do sistema. A tese apresenta os fundamentos teóricos de ambientes virtuais de educação, bem como das redes neurais artificiais SOM, utilizadas no modelo proposto. Também é apresentado o processo de desenvolvimento do sistema, a construção do treinamento automatizado e a estruturação do sistema tutor. A transmissão de conhecimentos é inspirada na transposição didática de conteúdos, com unidades didáticas organizadas em níveis que tem como objetivo desenvolver diferentes competências. As análises das redes SOM indicaram que o treinamento automatizado foi capaz de treinar diversas redes e identificar a rede com melhor ordenação topológica. Também são apresentadas comparações entre os desempenhos dos estudantes quando submetidos a estudos nos sistemas com orientação puramente especialista e híbrido. Os resultados dessa avaliação indicam a viabilidade do modelo proposto, pois o sistema mostrou-se capaz de aprender com os estudantes e ajustar as estratégias de ensino do professor. Os aprendizes que estudaram no sistema ampliaram suas médias nas avaliações de aprendizado do sistema e o tutor híbrido foi capaz de tomar decisões que ampliaram a aceitação das indicações de estudos do tutor.

Palavras Chave: Sistema Tutor Inteligente, STI, *Self Organizing Maps*, Redes SOM, Mapas de Kohonen, STI baseado em Redes Neurais, STI Híbrido.

ABSTRACT

The contemporary education has a lot of challenges and among them is the adaptation of using new technologies with classical education paradigm. It hasn't been different with distance education. In this context, this work proposes to develop a hybrid tutoring system model with decisions based on the teachers knowledge and help from Self Organizing Maps (SOM) or Kohonen's Maps neural networks. The proposed system has a initial teaching method that is set by specialist teacher up and while system is being running this pedagogic method is refined by the neural networks, which use patterns extracted from students that has used the system. The model proposes the utilization of a basic neural network structure with automated training which is capable of train several networks and define the one which represents results that is more coherent with the pattern's set, dismissing the intervention of a specialist on the evaluation of the network training performance. The system has adaptive and reactive features related to the apprentice, being able to offer to the students a personalized and dynamic learning. The system was developed in a web environment aiming avail the advantages of this technology. At this work, besides the proposed model developing it also were performed a data gathering with fresh students from integrated learning technical of Federal Institution of Goiás, Luziânia, Goiás, Brazil, to evaluate system's applicability. This thesis presents the fundamentals theorists of the virtual education environment, as also the artificial neural networks SOM, used on proposed model. Likewise, it shows the system developing process, the automated training build, in addition with the system tutor structure. The knowledge's transmission is inspired in the content's didactic transposition, with organization didactic units in levels that aim develop distinct skills. The SOM networks analysis indicate that the automated train was able to train several networks and identify a network with best topologic order. Moreover, this work presents a comparison between students performance when submitted to learn using the system with purely specialized orientation and hybrid orientation. The outcomes of this evaluation points out the viability of the proposed model, since the system has shown to be able to learn from students and adapt the teacher learning method. The apprentices that studied utilizing the system had amplified theirs grades on the learning system evaluations and the hybrid tutor was capable of take decisions which magnify the acceptation of the tutor learning indication.

Keywords: Intelligent Tutoring Systems, ITS, Self Organizing Maps, SOM, Kohonen Maps, ITS based on neural networks, Hybrid ITS.

LISTA DE FIGURAS

Figura 2.1: Arquitetura tradicional de um STI, adaptado de Giraffa (1995).....	30
Figura 2.2: Arquitetura de um STI conexionista.	35
Figura 2.3: Arquitetura de STI proposta por Raabe e Giraffa, 2005	39
Figura 3.1 : Estrutura do neurônio humano (CARVALHO et al, 1998).	43
Figura 3.2: Estrutura do neurônio de MacCulloch e Pitts (CARVALHO et al, 1998).....	43
Figura 3.3: Córtex cerebral humano (SEELEY et al, 1996).....	46
Figura 3.4: Modelo do mapa de Kohonen ou rede SOM.....	47
Figura 3.5: Formato para a região de vizinhança de uma SOM linear em anel.	48
Figura 3.6: Vizinhança de uma rede SOM. a) hexagonal, b) regular.	49
Figura 3.7: Fases do treinamento de uma rede SOM (KOHONEN, 1997)	52
Figura 3.8: Algoritmo básico de treinamento de uma rede SOM.....	53
Figura 3.9: Matriz com pesos associados aos neurônios de uma rede SOM.....	55
Figura 3.10: Visualização de pesos associados aos neurônios de uma rede SOM.....	55
Figura 3.11: Chapéu mexicano utilizado na atualização de pesos de uma rede SOM.	57
Figura 4.1: Estrutura dos tutores especialista e híbrido adaptada de Carvalho (2002).	73
Figura 4.2: Tela de texto do tutor proposto neste trabalho.	75
Figura 4.3: Tela de exercícios do tutor proposto neste trabalho.....	77
Figura 4.4: Navegador do tutor proposto neste trabalho.	78
Figura 4.5: Interface de cadastro de cursos no sistema proposto neste trabalho.	81
Figura 4.6: Tela de cadastro de conteúdos do modelo proposto neste trabalho.	83
Figura 4.7: Árvore de sugestão de decisão local.	84
Figura 4.8 Algoritmo básico do treinamento automatizado	91
Figura 4.9: Ilustração do processo de rotulação dos neurônios.....	93

Figura 4.10: Algoritmo de rotulação da rede SOM (para todos os neurônios).	94
Figura 4.11: Algoritmo de rotulação da rede SOM (apenas neurônios vencedores).....	96
Figura 5.1: Visitas por unidades didáticas e níveis.	103
Figura 5.2: Avaliação dos testes iniciais e finais nos tutores.	107
Figura 5.3: Desvio padrão dos testes iniciais e finais nos tutores.	108
Figura 5.4: Respostas a exercícios das unidades didáticas do tutor especialista.	110
Figura 5.5: Respostas a exercícios das unidades didáticas do tutor híbrido.....	111
Figura 5.6: Comparação da aceitação de indicação dos tutores.	113
Figura 5.7: Nível de satisfação dos tutores especialista e híbrido.	114

LISTA DE QUADROS E TABELAS

Tabela 3.1: Quadro de padrões a serem treinados no exemplo	64
Tabela 3.2: Quadro com vetor de pesos iniciais do exemplo	65
Tabela 3.3: Quadro do vetor de pesos após a primeira atualização.....	67
Tabela 4.1: Quadro de botões de resposta com níveis de satisfação.....	79
Tabela 5.1: Quadro de recursos computacionais utilizados neste trabalho	99
Tabela 5.2: Quadro das redes do treinamento automatizado.....	106

LISTA DE SIGLAS E ABREVIATURAS

CAI – Computed Aided Instruction ou Computed Assisted Instruction

IA – Inteligência Artificial

ICAI – Intelligent Computed Assisted Instruction

MLP – Multilayer Perceptron (Perceptron multcamadas)

RNA – Rede neural artificial

SOM – Self Organising Maps (Mapas Auto-Organizáveis)

STI – Sistemas Tutores Inteligentes

SUMÁRIO

CAPÍTULO 1: INTRODUÇÃO 15

1.1	SOBRE O TEMA	16
1.2	OBJETIVOS DESTE TRABALHO	17
1.3	METODOLOGIA	17
1.4	MOTIVAÇÃO	17
1.5	JUSTIFICATIVA	18
1.6	PROBLEMAS E HIPÓTESES	20
1.7	ESTRUTURA DO TRABALHO	21

CAPÍTULO 2: AMBIENTES VIRTUAIS DE EDUCAÇÃO 23

2.1	HISTÓRICO DA APLICAÇÃO DA INFORMÁTICA NA EDUCAÇÃO	24
2.2	INSTRUÇÃO ASSISTIDA POR COMPUTADOR (CAI)	26
2.3	SISTEMAS TUTORES INTELIGENTES	27
2.3.1	INSTRUÇÃO ASSISTIDA POR COMPUTADOR (CAI) VERSUS SISTEMAS TUTORES INTELIGENTES (ICAIs OU STIs)	28
2.4	ARQUITETURA TRADICIONAL DE UM STI	29
2.4.1	BASE DE DOMÍNIO	30
2.4.2	MODELO DO ALUNO	31
2.4.3	ESTRATÉGIAS DE ENSINO	31
2.4.4	MODELO DE INTERFACE	31
2.4.5	MÓDULO DE CONTROLE	32
2.5	ABORDAGEM DE STI UTILIZANDO AGENTE INTELIGENTE	32
2.6	STIs BASEADOS EM REDES NEURAIS ARTIFICIAIS	34
2.7	STIs UTILIZANDO APRENDIZAGEM POR REFORÇO	38
2.8	STI MEDIADO	38
2.9	ABORDAGENS UTILIZANDO ALGORITMOS GENÉTICOS	40

CAPÍTULO 3: REDES NEURAIS ARTIFICIAIS 41

3.1	SISTEMAS CONEXIONISTAS	42
3.2	MAPAS AUTO-ORGANIZÁVEIS	45

3.2.1	ESTRUTURA DA REDE SOM	47
3.2.2	TREINAMENTO	49
3.2.3	ALGORITMO DE TREINAMENTO	52
3.2.4	DETERMINAÇÃO DO NEURÔNIO VENCEDOR	53
3.2.5	AJUSTE DE PESOS	54
3.2.6	COMO CONSTRUIR BONS MAPAS	57
3.2.7	MÉTRICAS DE QUALIDADE	61
3.2.8	EXEMPLO DE UMA REDE NEURAL SOM	63

CAPÍTULO 4: MODELO PROPOSTO **69**

4.1	AMBIENTE TECNOLÓGICO DO SISTEMA	70
4.2	ESTRUTURA DOS TUTORES COM DECISÃO ESPECIALISTA E HÍBRIDO	72
4.3	O TUTOR COM DECISÃO ESPECIALISTA	79
4.3.1	DESLOCAMENTO DO EIXO ENTRE OS CONTEÚDOS	80
4.3.2	DESLOCAMENTO ENTRE OS NÍVEIS DE UM MESMO CONTEÚDO	82
4.4	O TUTOR HÍBRIDO	85
4.4.1	ESTRUTURA DO TUTOR COM DECISÃO HÍBRIDA	85
4.4.2	ENTRADAS DAS REDES NEURAIS	88
4.4.3	O TREINAMENTO AUTOMATIZADO DA REDE	89
4.4.4	ROTULAÇÃO DAS REDES NEURAIS	92

CAPÍTULO 5: RESULTADOS OBTIDOS **97**

5.1	REALIZAÇÃO DO EXPERIMENTO	97
5.1.1	FORMATAÇÃO DO CONTEÚDO	98
5.1.2	RECURSOS COMPUTACIONAIS UTILIZADOS NESTE TRABALHO	99
5.1.3	DEFINIÇÃO DAS REGRAS ESPECIALISTAS	100
5.1.4	COLETA DE DADOS NOS TUTORES ESPECIALISTA E HÍBRIDO	100
5.1.5	CONJUNTO DE TREINAMENTO	101
5.1.6	CONJUNTO DE VALIDAÇÃO	104
5.2	AVALIAÇÃO DO TREINAMENTO AUTOMATIZADO	104
5.3	ANÁLISE DOS RESULTADOS DOS TUTORES	106

CAPÍTULO 6: CONCLUSÕES E CONTRIBUIÇÕES DESTE TRABALHO E TRABALHOS FUTUROS **116**

6.1 CONCLUSÕES	116
6.2 CONTRIBUIÇÕES	118
6.3 TRABALHOS FUTUROS	120
6.4 PUBLICAÇÕES DO AUTOR RELACIONADAS A ESTE TRABALHO	120
6.5 PUBLICAÇÕES SUBMETIDAS E EM REVISÃO	122
<u>REFERÊNCIAS</u>	124

CAPÍTULO 1: INTRODUÇÃO

Ensinar não é uma função vital, porque não tem o fim em si mesma; a função vital é aprender. (Aristóteles)

Diversas áreas têm trabalhado em pesquisas de novas tecnologias, modelos e metodologias para serem aplicados na educação a distância e em sistemas de ensino computadorizados. Durante muitas décadas, estudos sobre Sistemas Tutores Inteligentes (STI) têm sido apresentados em Engenharia da Computação, principalmente na área de Inteligência Artificial (IA). Para ser considerado inteligente, um STI deve ter um comportamento semelhante ao tutor humano e ser capaz de oferecer um ensino adaptativo, reativo, flexível e personalizado (GIRAFFA E VICCARI, 1997).

A Inteligência Artificial tem pesquisado STI em diversas técnicas e abordagens. STI utilizando paradigmas simbólicos da IA, com mediação do professor, apresentaram resultados promissores (RAABE E GIRAFFA, 2006). Paradigmas conexionistas e não lineares também têm sido usados no desenvolvimento de Sistemas Tutores Inteligentes (MARKOWSKA-KACZMAR et al, 2010).

Neste trabalho são descritas as etapas de desenvolvimento de um modelo de sistema tutor inteligente híbrido, com orientação do professor especialista, subsidiadas por redes neurais artificiais. São apresentados os resultados obtidos no

estudo e realizada uma comparação entre o comportamento dos modelos com orientação especialista e híbrida, ambos desenvolvidos para esta pesquisa.

1.1 Sobre o Tema

Nos últimos anos a comunidade educacional passou a entender as vantagens dos cursos que têm sua base em tecnologia da informação. O computador tem sido empregado tanto no treinamento no âmbito presencial, quanto no ensino presencial, semipresencial e a distância. Universidades virtuais são hoje uma realidade, com propostas de ensino em níveis de treinamento e sistemas educacionais (RAABE E GIRAFFA, 2006).

A Inteligência Artificial tem estudado a construção de sistemas tutores inteligentes sob diversas técnicas e abordagens (VICCARI E GIRAFFA, 1996). Sistemas baseados em agentes têm se apresentado como uma alternativa viável na construção de Sistemas Tutores Inteligentes (DORÇA, 2003). Pesquisas nessa área buscam prover um ensino adaptativo, contextualizado, direcionado e flexível.

As redes neurais Self Organizing Maps (SOM) foram aplicadas na construção de STI por Carvalho (2002) e também utilizadas por Dorça et al (2009) para classificar o estudante quanto à adequação do conteúdo didático.

A utilização de técnicas como cadeias de Markov e Algoritmos Genéticos também têm sido aplicadas na construção de sistemas adaptativos ao perfil do estudante (DORÇA et al, 2012).

O tema deste trabalho está inserido nesse contexto, reforçando a atua-

ção da Engenharia da Computação na construção de modelos que possam auxiliar na aplicabilidade da tecnologia da informação no campo educacional.

1.2 Objetivos deste trabalho

O objetivo geral deste trabalho consiste em desenvolver um modelo de Sistema Tutor Inteligente Híbrido, adaptativo ao aprendiz, capaz de orientar o estudante, combinando o conhecimento do professor especialista e a decisão de uma rede neural artificial.

Como objetivos específicos têm-se o projeto e a construção de um sistema tutor com orientação puramente especialista, para a coleta de dados, e a construção de um modelo híbrido em que suas decisões tem por base o conhecimento especialista e as redes neurais SOM, além da coleta de dados utilizando o sistema tutor híbrido.

1.3 Metodologia

Neste trabalho é realizada a pesquisa bibliográfica sobre os fundamentos teóricos de STI e das redes neurais, a pesquisa de campo para a coleta de dados e a avaliação dos dados coletados no sistema proposto.

1.4 Motivação

A evolução do conhecimento propiciou à humanidade vivenciar experiências antes não imaginadas, promovidas, em grande parte, pela evolução tecnológica. Em especial, a tecnologia da informação tornou-se indispensável para a gestão

da vida moderna e está inserida nas organizações, residências, governo e também na educação. Neste último caso ainda é comum manter as estruturas educacionais clássicas que consistem apenas da sala de aula e do professor, mesmo já tendo sido incorporados muitos aspectos tecnológicos no processo de ensino aprendizagem, tais como laboratórios de informática, salas multimídia, dentre outros.

Com a perspectiva de poder contribuir de forma técnica e científica para a construção de sistemas tutores inteligentes, a atuação do autor deste trabalho na área educacional e sua formação em computação foram fatores motivadores para a realização deste trabalho.

1.5 Justificativa

Desde a década de cinquenta do século passado o computador tem sido utilizado na Educação como ferramenta auxiliar na tarefa de ensino e aprendizagem. Os primeiros sistemas computacionais aplicados na educação são classificados como Instrução Assistida por Computador ou Sistemas CAI (CAI, do inglês *Computer-Assisted Instruction*). Em tais sistemas, os métodos educacionais muitas vezes são apresentados de modo expositivo e centrados no professor. Com a finalidade de reforçar o aprendizado, o estudante deve antes compreender a lição apresentada e depois responder aos exercícios no computador (GIRAFFA E VICCARI, 1997).

Com o avanço das técnicas de Inteligência Artificial, nasceu neste meio um novo paradigma de sistemas educacionais, denominados ICAI (do inglês *Intelligent Computer-Assisted Instruction*) ou Sistemas Tutores Inteligentes (STI).

Nesta nova geração os sistemas passaram a ter a capacidade de representar determinados domínios e de se adaptar às características dos estudantes. Os sistemas ICAI apresentam uma ampla representação de seu domínio, permitindo utilizar seus conhecimentos de maneira não diretamente explicitados pelo projetista.

A partir dos anos 2000 os métodos conexionistas passaram a ser aplicados na construção de STI, com o objetivo de construir modelos cujas regras tivessem suas origens na forma humana de agir ou estudar.

Diante do contexto atual dos projetos de STI, justifica-se explorar a combinação de técnicas que possam incluir o conhecimento do professor, na definição das regras iniciais, e as decisões neurais, para auxiliar na orientação do estudo. Neste trabalho o sistema é treinado com dados extraídos dos próprios alunos, com a finalidade de permitir orientações de navegação baseadas nas suas formas de estudar.

Esta tese utiliza as redes neurais denominadas Mapas Auto-Organizáveis ou *Self Organizing Maps* devido à sua capacidade de manter uma ordenação topológica dos padrões treinados. Acredita-se que as pessoas tendem a manter um padrão de respostas e navegação quando submetidas a determinadas tarefas. Assim, as redes neurais SOM auxiliam no ajuste da decisão do professor, refinando-a e tornando o sistema adaptativo ao perfil de navegação do estudante, com o objetivo de permitir um ensino individualizado e personalizado. O trabalho justifica-se ainda em virtude da necessidade de novas soluções para utilizar a informática na sala de aula.

1.6 Problemas e hipóteses

O sistema educacional tem presenciado o enorme crescimento das tecnologias da informação e admitido o computador como potencial recurso de ensino. Vários projetos têm sido voltados para essa área, como a criação de cursos presenciais, realizados por meio de computadores, e até mesmo cursos a distância, via Internet (ALMEIDA, 2006).

A popularização da Internet proporcionou mais facilidades e oportunidades para novos projetos. Além de possibilitar a troca de informações de forma rápida e dinâmica, ela tem proporcionado o rompimento de fronteiras geográficas, permitindo ainda ampliar as opções de ensino a distância.

Desde 1840, a sociedade tem experimentado os benefícios que o ensino a distância oferece. Inicialmente, os meios de correspondência eram utilizados para distribuir o material didático impresso e os alunos desenvolviam seus estudos de acordo com a sua disponibilidade de tempo (HORTON, 2000).

Com a adoção de novas tecnologias e com a utilização das redes de computadores, o ensino passou a agregar recursos inovadores. O novo modelo também trouxe a necessidade de profundas mudanças na forma de ensinar.

Entender os benefícios que as novas tecnologias oferecem não é uma tarefa fácil e exige o trabalho de equipes inter e multidisciplinares, constituídas de profissionais da Educação e da Tecnologia, entre outros.

A maioria dos sistemas de ensino a distância disponíveis não utilizam inteligência artificial em seus projetos. As modalidades a distância mais comuns

limitam-se ao uso de plataformas educacionais que disponibilizam fóruns, bate-papos, diário de bordo, disponibilização de conteúdos, entre outros, sem a integração da Inteligência Artificial que as tornem reativas ao aprendiz. Em geral, esses sistemas não oferecem um ensino adaptativo e não consideram os diferentes perfis dos estudantes, para orientá-los no seu processo de estudo. Apesar de algumas propostas utilizarem recursos para promover maior interação e individualização, em geral eles mantêm sua característica comum de distribuição de conteúdo digital estático e sequencial (GIRAFFA, 1995).

A hipótese básica é de que o modelo proposto neste trabalho será capaz de aprender com base na estratégia de estudo do aluno. Como hipótese secundária, supõe-se que o modelo será capaz de aperfeiçoar as decisões do especialista e reestruturar a estratégia de ensino do tutor, tornando-se mais adaptativo ao perfil de estudo do aluno.

1.7 Estrutura do trabalho

O segundo capítulo, apresenta as fundamentações teóricas dos ambientes virtuais da educação, desde os primeiros Sistemas CAI até as diferentes abordagens de Sistemas ICAI ou Sistemas Tutores Inteligentes (STI). É descrita a estrutura básica do STI tradicional, bem como as principais abordagens conexionistas.

O terceiro capítulo, fundamenta a abordagem conexionista de Inteligência Artificial ou, simplesmente, redes neurais artificiais (RNAs). São apresentadas mais detalhadamente as redes neurais *Self Organizing Maps* (SOM) ou Mapas Auto-Organizáveis, pois elas são utilizadas neste trabalho. Também são apresenta-

das as orientações para a criação de redes SOM mais bem elaboradas e as técnicas para mensurar o aprendizado dessas redes. Finalmente, é mostrado um exemplo simples de uma rede desse tipo.

O quarto capítulo descreve o modelo proposto incluindo o sistema com orientação especialista, desenvolvido para a coleta de dados, e o sistema tutor híbrido. Ainda são mostradas as entradas das redes neurais que compõem o conjunto de treinamento e são especificados os critérios adotados para o treinamento automatizado.

O quinto capítulo mostra os experimentos e os resultados obtidos nos testes realizados, incluindo a coleta de dados, a composição do conjunto de treinamentos, as características das redes neurais utilizadas no tutor e o processo de criação dos rótulos dessas redes. Também são apresentados os resultados da comparação entre o sistema utilizado na coleta de dados, com orientação especialista, e o modelo híbrido proposto.

O capítulo seis apresenta as conclusões, as principais contribuições e as publicações relacionadas a este trabalho, bem como os trabalhos futuros que poderão ser desenvolvidos a partir desta tese.

Assim, espera-se que os resultados deste trabalho possam trazer aos pesquisadores, interessados nesta área de conhecimento, uma visão da aplicabilidade dos modelos híbridos na construção de Sistemas Tutores Inteligentes (STI) que integrem o conhecimento do professor especialista e a forma de estudo dos alunos, no processo de transposição didática de conteúdos.

CAPÍTULO 2: AMBIENTES VIRTUAIS DE EDUCAÇÃO

O educador lida com a arte de educar. O instrumento de sua arte é a Pedagogia. Ciência da Educação, do ensinar.

*É no seu ensinar que se dá seu aprendizado de artista...
... No exercício disciplinado de sua arte (mediado por seus instrumentos metodológicos), é que a paixão de educador é educada.
(Madalena Freire)*

Durante décadas tem-se desenvolvido pesquisas sobre a aplicação de tecnologia no ensino. Áreas como a Educação, Engenharia, Computação e Psicologia, entre outras, têm contribuído para a evolução de modelos cada vez mais bem elaborados.

A utilização de ambientes virtuais na educação apresentou-se em vários formatos e tornou-se mais evidente a partir da criação da web, na década de 1990. No final dessa década e início dos anos 2000 surgiram as propostas de Treinamento Baseado na Web, do inglês *Web Based Training* (WBT). Com igual intensidade, foram as promessas de sistemas tutores *online* que propiciariam a democratização do ensino por meio da educação a distância. Todavia, o desenvolvimento de sistemas tutores em ambientes virtuais é uma tarefa complexa e consiste de conhecimentos multidisciplinares e transdisciplinares, fator limitador para que a expansão esperada fosse atingida.

Neste contexto, este capítulo apresenta o histórico da aplicação de ambientes virtuais na educação e os principais modelos desenvolvidos e aplicados

nesses ambientes de ensino.

2.1 Histórico da aplicação da informática na educação

A utilização da informática na educação não é recente e pesquisas têm sido desenvolvidas desde a década de 1950. Foi nesta época que os primeiros sistemas conhecidos como CAI (do inglês *Computed Aided Instruction ou Computed Assisted Instruction*) começaram a ser projetados. A evolução de tais sistemas recebeu a denominação de ICAIs (do inglês *Intelligent Computed Assisted Instruction*) ou Sistemas Tutores Inteligentes (STIs), por utilizarem conceitos de Inteligência Artificial em sua concepção (GIRAFFA E VICCARI, 1997).

O computador tem se apresentado como um grande aliado no auxílio ao ensino e várias são as formas de sua aplicação. Geralmente, são utilizados softwares educacionais, que são programas desenvolvidos no intuito de auxiliar no processo de ensino aprendizagem. Todavia, quaisquer outros softwares podem ser adotados para este fim, desde que uma metodologia adequada seja utilizada (GIRAFFA E VICCARI, 1997).

Os *softwares* educacionais também podem ser classificados de várias formas de acordo com sua característica e aplicação. Segundo Taylor (1980), eles podem ser divididos em Tutor, Tutelado e Ferramenta.

Lucena (1994) considerou ainda que devido à incorporação de novas tecnologias e ao avanço de recursos de hardware e de software, a classificação de softwares educacionais pode ser ampliada para Tutor, Tutelado, Ferramenta, Exer-

cício e Prática, Simulação e Jogos Educativos.

O conceito de Tutor aplica-se ao software onde o computador assume o papel de um “professor”. Nesse caso, o software possui a capacidade de apresentar os conceitos ao estudante e oferecer níveis diferenciados de acordo com as avaliações das respostas obtidas do educando. Segundo Giraffa (1995), nessa modalidade encontram-se alguns sistemas CAIs, onde o modelo de ensino aprendizagem é concebido unicamente em instrução programada com apoio na teoria comportamentalista de Skinner (PARK; PEREZ; SEIDEL, 1987).

Na classificação Tutelado, o estudante é quem instrui o computador a fazer determinadas tarefas. Ele pode ser exposto a uma linguagem de programação, por exemplo LOGO, e deve programar um computador.

A definição de Ferramenta é atribuída quando o computador é utilizado como um auxiliar no processo de ensino aprendizagem, em que editores de textos, planilhas eletrônicas, calculadoras, programas gráficos etc. são usados para esse fim.

O conceito de Exercício e Prática ocorre quando os exercícios são gerados pelo computador, considerando a formação do estudante, com a finalidade de testar e fixar o conhecimento em áreas específicas.

Por sua vez, Simulação é a definição atribuída a um tipo específico de software em que o computador é utilizado para imitar um sistema real ou imaginário.

Por fim, na especificação de Jogos Educativos o computador é utiliza-

do de forma lúdica para desenvolver um determinado ramo de conhecimento do estudante.

Uma das dificuldades encontradas para a criação de softwares educacionais, sejam eles CAI ou ICAIs, está na capacidade de reunir uma equipe inter e multidisciplinar que possibilite um bom resultado técnico e pedagógico. Na criação desses sistemas, não se pode desprezar a necessidade da presença de especialistas da área da Computação, Engenharia de Software, Pedagogia, Psicologia, Designer Gráfico, entre outras (GIRAFFA, 1995).

Para maior compreensão sobre os conceitos de softwares aplicados na educação, a seguir são apresentados com maior detalhamento as concepções de CAI e ICAI.

2.2 Instrução Assistida por Computador (CAI)

Instituição Assistida por Computador ou Sistemas CAI tiveram sua origem na área educacional, influenciada pela Teoria Comportamentalista de Skinner. Sua abordagem é centrada no professor e sua finalidade é auxiliar o professor no processo de ensino (PARK; PEREZ; SEIDEL, 1987). Nesse modelo, inicialmente o aluno deve receber explicações expositivas para depois exercitá-las no computador, resolvendo exercícios, por exemplo.

Um dos principais problemas dos sistemas CAI é que eles não levam em consideração as diferenças existentes entre os alunos. Nesses formatos, o conteúdo é previamente elaborado e consiste, muitas vezes, em um simples “*virador de*

“páginas”, ou seja, uma extensão dos livros de papel, agora digitalizados (GIRAFFA, 1995).

Nos sistemas CAI, o perfil do estudante é ignorado e a exposição do conteúdo é idêntica para todos os educandos, independente de um estudante ter um desenvolvimento superior ou inferior ao de outro. Com a aplicação de Inteligência Artificial, os sistemas receberam a denominação de ICAI, cujos conceitos são apresentados a seguir.

2.3 Sistemas Tutores Inteligentes

Na década de 1970 passou-se a utilizar as técnicas de Inteligência Artificial (IA) nos sistemas CAI, originando os sistemas denominados *Intelligent Computer Assisted Instruction* (ICAI) ou Sistemas Tutores Inteligentes (STIs). Procurou-se criar programas que deixassem de ser meros “*viradores de páginas*”, atribuindo-lhes características adaptativas (GIRAFFA, 1995).

Os sistemas ICAIs tiveram sua origem na área da Computação e possuem sua base teórica na Psicologia Cognitivista. Sua estrutura básica é subdividida em módulos e sua sequência de execução depende da resposta do aluno. A estruturação do conhecimento é heurística e a modelagem do aluno tenta avaliar todas as respostas durante a interação com o sistema.

Com a utilização de técnicas de IA, os sistemas ICAIs são capazes de analisar os padrões de erros, o estilo e a capacidade de aprendizagem do aluno. Assim, eles permitem oferecer instrução especial sobre o conceito que o aluno está

apresentando dificuldade (SAVIANI, 1991).

Um tutor para ser considerado inteligente deve ser flexível. Isto significa que ele deve ter capacidade para aprender sobre o contexto atual (meio) e atuarizar o seu conhecimento (VICCARI, 1990). Neste tipo de software, o aluno aprende fazendo e, igualmente, o sistema adapta-se ao desenvolvimento do aluno. Para melhor compreensão, é apresentada a seguir uma comparação entre os sistemas CAI e ICAI.

2.3.1 Instrução Assistida por Computador (CAI) versus Sistemas Tutores Inteligentes (ICAIs ou STIs)

As diferenças fundamentais entre os ICAIs e os CAIs tradicionais estão na forma pelas quais seus projetos são concebidos (GIRAFFA, 1995). Em sistemas CAI, os alunos são induzidos a uma resposta correta, baseada em estímulos previamente planejados. Os ICAIs buscam extrair capacidades cognitivas do aluno e utilizar esses resultados para a tomada de decisão.

Uma das maiores críticas aos sistemas CAIs reside em sua incapacidade de manter um diálogo aberto com o aluno. Nos sistemas ICAIs, técnicas de IA são aplicadas de forma a proporcionar instruções adaptadas, sejam elas no sentido de contexto ou forma.

De fato, os STIs deveriam assumir um papel mais próximo do professor humano. Todavia, isso está longe de acontecer em virtude das numerosas limitações, tanto de software quanto de hardware. Os seres humanos têm a possibilidade

de trabalhar com diversos aspectos sensoriais como olfato, tato, visão etc. e analisar as respostas a esses estímulos. Essa capacidade sensorial ainda é inviável do ponto de vista tecnológico.

Para Jonassen e Wang (1993), um STI deve passar em três testes antes de ser considerado “inteligente”: Disposição do Conteúdo, Capacidade de Avaliação e Estratégias de Ensino.

Para a Disposição do Conteúdo, segundo Jonassen e Wang (1993), o tema ou a especialidade estudada deve ser codificada de modo que o sistema possa acessar às informações, fazer inferências ou resolver problemas. Com referência à Capacidade de Avaliação, o sistema deve ser capaz de avaliar a aquisição de conhecimento pelo estudante; e, por fim, com relação às Estratégias de Ensino do tutor, estas devem ser projetadas para reduzir a discrepância entre o conhecimento do especialista e o conhecimento do estudante.

Para melhor compreensão do processo de construção de um STI, a seção 2.4 deste capítulo apresenta a arquitetura básica de um tutor que utiliza o paradigma da Inteligência Artificial Simbólica (IA Simbólica) na sua concepção.

2.4 Arquitetura Tradicional de um STI

Viccari e Giraffa (1996) mencionaram que as arquiteturas dos sistemas tutores inteligentes variam de uma implementação para outra. Mas que, em geral, os STIs tradicionais possuem uma organização básica, como mostrada na Figura 2.1.

Cada um dos componentes de um STI deve ser objeto de reflexão por parte dos engenheiros do conhecimento que participam do processo de sua construção. A seguir são apresentadas descrições dos componentes básicos de um STI.

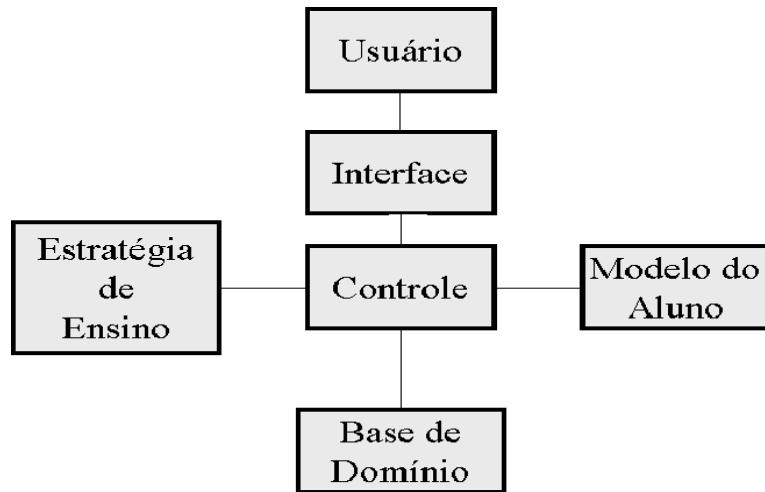


Figura 2.1: Arquitetura tradicional de um STI, adaptado de Giraffa (1995).

2.4.1 Base de domínio

A base de domínio é o componente que age como o especialista e desempenha a função da fonte de conhecimento a ser apresentada. Ela é considerada como um padrão para as avaliações de desempenho do estudante e constituída pelo material instrucional, por um sistema de geração de exemplos, por processos de simulação e pela possibilidade de formulação de diagnósticos (GIRAFFA, 1995). Em essência, a base de domínio contém a maior parte da “inteligência” do sistema na forma do conhecimento necessário para solucionar problemas do domínio (PARK, 1988). O grande desafio é fornecer uma rica representação desse domínio, de forma que possibilite flexibilidade no ensino.

2.4.2 Modelo do aluno

O conhecimento que o sistema deve ter do seu próprio usuário proporcionará um ensino personalizado e inteligente. O modelo do aluno representa o conhecimento e as habilidades cognitivas do usuário em um determinado momento. A característica principal deste modelo é a de tentar contemplar todos os aspectos do conhecimento e do comportamento do estudante que tragam consequências para o seu desempenho e aprendizagem (ALENCAR, 2000).

2.4.3 Estratégias de ensino

As estratégias de ensino constituem o modelo pedagógico em si e podem ser vistas como planos de apresentação do material instrucional. Esse modelo deve conter o conhecimento necessário para a tomada de decisões e sobre quais técnicas de ensino empregar. A construção dessas estratégias é bastante complexa e exige muita versatilidade, pois, as decisões devem ser tomadas dinamicamente, tendo como base as atitudes e as necessidades do aluno.

2.4.4 Modelo de interface

De modo geral pode-se afirmar que o usuário tem a concepção de que a interface é o próprio sistema. Logo, fica evidente que uma atenção especial deve ser dispensada nesse sentido. A interface é a responsável pela apresentação do material instrucional e pela manipulação do progresso do estudante, por meio da recepção da resposta do aluno. Para esse contexto, Giraffa (1995) menciona que devem ser considerados a agradabilidade, a facilidade, o tempo de resposta e a monitora-

ção.

A agradabilidade é necessária para evitar que o estudante fique entediado, ou seja, é preciso riqueza de recursos na apresentação do material instrucional. A facilidade é desejável para a troca de iniciativa do diálogo. O estudante deve ter a possibilidade de intervir facilmente no discurso do tutor. O tempo de resposta deve, evidentemente, permanecer dentro dos limites aceitáveis e a monitoração deve ser realizada o máximo possível em “*background*”, para não onerar o estudante com excessivos questionários, mas respeitando também a barreira do tempo e de responsa.

2.4.5 Módulo de controle

O módulo de controle é o responsável pela coordenação geral do tutor. Geralmente são as funções programadas para trocas de informações, apresentação de interfaces etc. (ALENCAR, 2000). Esse módulo consiste de funções tais como acesso a banco de dados, manutenção do histórico da sessão do usuário e comunicação com outros programas utilitários que façam parte do STI.

A abordagem vista nesta subseção refere-se ao modelo tradicional de STI. Porém, sistemas que utilizam agentes inteligentes, modelos conexionistas, entre outros, também têm sido desenvolvidos e são apresentados a seguir.

2.5 Abordagem de STI Utilizando Agente Inteligente

Outro modo de formular um STI é considerando uma abordagem de sistema cooperativo multi-agentes. Um agente é definido como um sistema capaz de

perceber por meio de sensores e agir em um determinado ambiente (ALENCAR, 2000).

Os agentes são frequentemente considerados como uma entidade capaz de exibir alguns aspectos da inteligência humana. Em geral, os agentes funcionam contínua e autonomamente em um ambiente no qual existem outros processos e agentes (CORREA FILHO, 1994). Os agentes também podem ser vistos como entidades às quais se atribuem estados, denominados de estados mentais, que usualmente podem ser crenças, decisões, capacidades, objetivos, intenções, compromissos e expectativas, conceitos análogos ou similares aos seres humanos (SHOHAM, 1993).

Desse modo, podem existir diversos agentes no sistema, cada um com sua função. Eles podem agir, por exemplo, modelando o aluno, o sistema tutor e/ou a base de conhecimento. A interação entre esses agentes é que determina o comportamento “inteligente” do sistema tutor.

Dorça, Lopes e Fernandes (2003) apresentaram uma arquitetura multi-agente de STI para sistemas de educação a distância. O modelo possuía características adaptativas e a arquitetura proposta foi desenvolvida para sistemas de educação baseados na web. Seu desenvolvimento consistiu da construção de dois modelos: o primeiro deles similar aos modelos tradicionais utilizados em treinamento baseado na web e, em um segundo momento, foi construído um sistema inteligente multi-agente que foi integrado ao modelo tradicional. A estrutura proposta tornou possível a participação do estudante no processo de aprendizagem e na construção do pró-

prio conhecimento do sistema.

Na busca por modelos inteligentes capazes de reagir às ações do aprendiz, ou seja, adaptar-se ao comportamento do estudante, e cujo desenvolvimento e a elaboração da estratégia de ensino tenha menor intervenção do conhecimento do especialista, foram criados STIs utilizando Redes Neurais Artificiais. Esses modelos são apresentados na seção 2.6 deste capítulo.

2.6 STIs Baseados em Redes Neurais Artificiais

No início dos anos 2000, foram apresentadas investigações sobre a aplicação de redes neurais artificiais em sistemas tutores inteligentes. Alencar (2000) demonstrou a capacidade de uma rede neural Perceptron multicamadas extrair padrões que poderiam ser utilizados para auxiliar na navegação de um sistema tutor inteligente.

Em 2002 Carvalho (2002) construiu um STI utilizando mapas de Kohonen. Pouco tempo depois, Melo et al (2005) também apresentaram um modelo que utilizava Perceptron Multi Camadas, do inglês *Multi Layer Perceptron* (MLP), e decisão probabilística na tomada de decisão. Nesse último modelo, Melo ainda utilizou padrões de treinamento extraídos das características psicológicas dos estudantes. Ambos os modelos apresentaram resultados satisfatórios.

Carvalho (2002) e Melo et al (2005) buscaram a criação de um sistema que pudesse tomar decisões independentemente da interferência de um especialista. Na maioria dos sistemas tutores encontrados até então, o trabalho

de um especialista era exigido para criar a Base de Domínio, responsável pelas tomadas de decisões (VICCARI E GIRAFFA, 1996).

A Figura 2.2 mostra a estrutura básica do modelo proposto por Carvalho (2002) e também utilizado por Melo et al (2005).

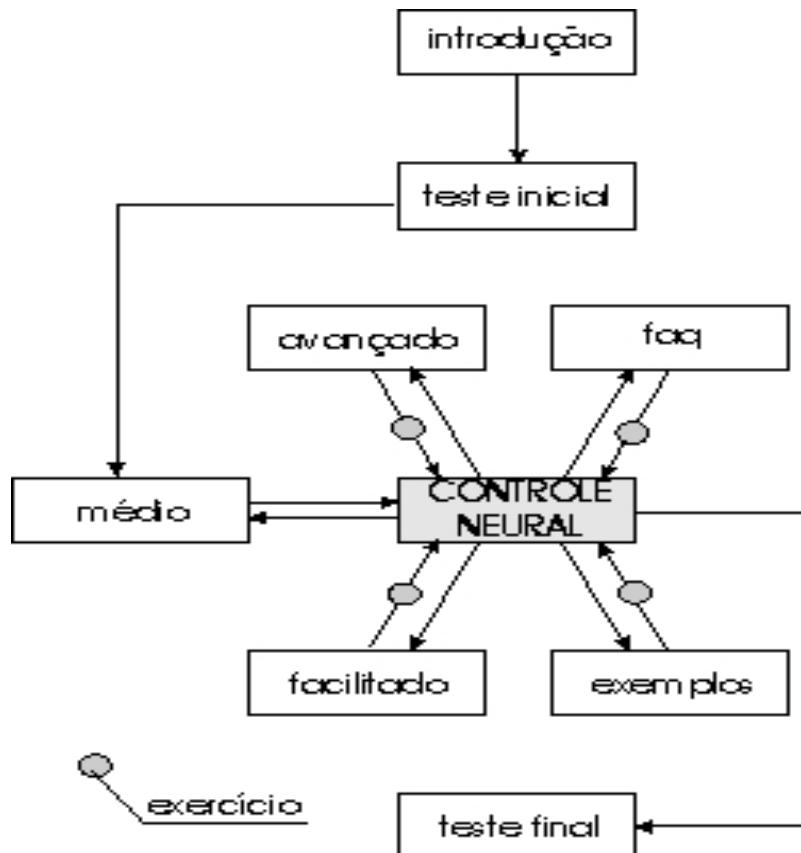


Figura 2.2: Arquitetura de um STI conexionista.

O modelo conexionista formula um tutor multinível, cuja adaptação ao perfil do aluno é realizada com base na forma com que as pessoas estudam ou navegam pelo ambiente virtual. Isso lhe atribui características de maior proximidade da forma humana de agir. Nos modelos conexionistas apresentados aqui, a estratégia de ensino é substituída pelas redes neurais artificiais, que passam a ser as responsáveis pelas tomadas de decisões.

Nos modelos conexionistas, o treinamento das redes neurais é realizado a partir de um conjunto de exemplos extraídos da coleta de dados utilizando um sistema denominado Tutor Livre, cuja estrutura é a mesma apresentada na Figura 2.2. No entanto, no lugar da decisão neural, existe um navegador com opções de escolha.

No Tutor Livre, o aprendiz deve responder a um teste inicial de onde então é submetido, invariavelmente, ao nível médio de estudo do primeiro conteúdo do curso. Após estudado o nível médio, o aluno é então encaminhado a um navegador que lhe proporciona autonomia e liberdade de escolha, a seu critério e sem nenhuma interferência especialista. No navegador, o estudante pode definir qual é o próximo passo, incluindo um nível diferente dentro do conteúdo atual, ou, se desejar, prosseguir para um novo conteúdo. No último caso, o aluno sempre será submetido ao nível médio do próximo conteúdo.

No modelo neural, o navegador é substituído pelo controle neural. A decisão neural retira do estudante o poder de escolha e passa a tomar as decisões de forma automatizada.

Em 2006, Almeida (2006) apresentou o Sistema Tutor Monitorado por em que uma rede neural é responsável por monitorar a navegação do estudante, durante todo o processo de aprendizado, e sugerir caminhos alternativos buscando o melhor desempenho. Todavia, caso o aprendiz decidisse por um caminho alternativo diferente da orientação do tutor e então obtivesse sucesso no teste final, o sistema se adaptava ao novo comportamento apresentado pelo aluno, incluindo-o no

treinamento do modelo neural. Esse sistema foi utilizado por um grupo de estudantes do ensino técnico de nível médio e os resultados apresentaram uma melhora de dez por cento no índice de aprovação da disciplina de linguagem de programação.

Mais tarde, em 2009, também foi apresentado por Dorça et al (2009) um sistema adaptativo ao aprendiz utilizando uma rede neural SOM para classificar o estudante quanto à adequação do conteúdo didático. Nessa época, Dorça et al (2009) consideraram para a tarefa de classificação do estudante e consequente escolha do conteúdo, o grau de acerto de respostas a testes, o tempo gasto para a resposta, o tempo gasto nas páginas de conteúdos, a quantidade de operações de rolagem das páginas e a participação do aluno em bate-papos (*chats*) ou *fóruns*.

Em 2010, Markowska-Kaczmar et al (2010) apresentaram uma pesquisa sobre as principais técnicas utilizadas na construção de sistemas de e-learning com capacidade adaptativa, entre essas está a aplicação de agrupamentos utilizando a rede SOM.

Melo et al (2012) apresentaram a formalização matemática para a construção de um modelo neural por padrões proximais de aprendizagem para automação personalizada de conteúdos didáticos.

Apesar dos bons resultados apresentados, os modelos conexionistas dependem de trabalho com a coleta de dados para o treinamento das redes neurais artificiais. Além disso, para cada curso construído também era necessária a interface de um engenheiro do conhecimento na formatação da estrutura da rede neural, o que dificultava a sua utilização (CARVALHO, 2002).

No intuito de eliminar o trabalho com a coleta de dados para treinamento, bem como o ajuste das redes neurais por um especialista, Martins e Afonseca (2007) propuseram a substituição das redes neurais por Aprendizado por Reforço, como mostrado na seção 2.7 deste capítulo.

2.7 STIs Utilizando Aprendizagem por Reforço

Em 2007, Martins e Afonseca (2007) apresentaram um modelo de Sistema Tutor Inteligente que tem como base a estrutura da Figura 2.2, cujas decisões utilizavam o paradigma denominado Aprendizagem por Reforço. Nesse modelo, a aprendizagem para a tomada de decisão ocorre com a utilização do próprio tutor, pelos estudantes. Durante sua utilização, o sistema se adapta à melhor forma, a partir da geração de estímulos positivos (reforço positivo) ou negativos (reforço negativo). Embora o sistema tenha mostrado bons resultados, ele começava a ser utilizado sem que nenhum conhecimento prévio estivesse disponível.

Outras soluções de STI passaram por adaptações de modelos tradicionais. A seção 2.8 deste capítulo apresenta uma variação do modelo clássico de STI, onde é incluída a mediação do professor.

2.8 STI Mediado

Em 2006, Raabe e Giraffa (2006) apresentaram uma alternativa para o modelo tradicional de ICAI em que seu desenvolvimento tinha como base, além da estrutura tradicional de STI, a aplicação da Teoria das Experiências de Aprendizagem Mediadas. Nesse modelo, o professor torna-se parte ativa do processo de deci-

são, uma vez que suas ações são incluídas no modelo do tutor. Nessa proposta, como pode ser observado na Figura 2.3, o tutor considera, além do perfil do aluno, as contribuições e intervenções do professor.

O *Intelligent Teaching Assistants* (ITA) possibilita a promoção de experiências de Aprendizagem Mediadas, segundo a teoria de Reuven Feuerstein (FEUERSTEIN, 1997 apud RAABE E GIRAFFA, 2006). Nessa proposta, além do Modelo do Aluno, inclui-se também o módulo do professor, onde as análises e monitoramentos auxiliam na decisão do sistema.

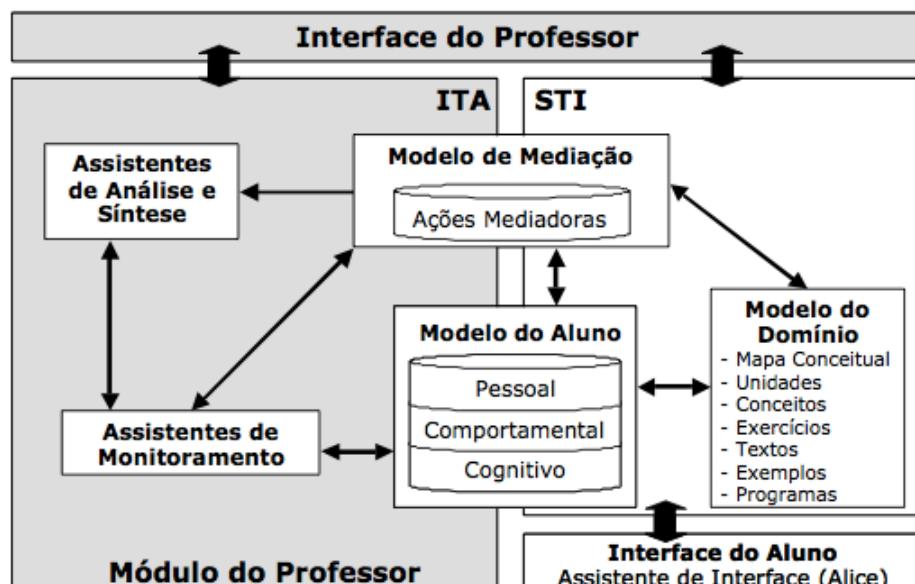


Figura 2.3: Arquitetura de STI proposta por Raabe e Giraffa, 2005

Com os trabalhos apresentados neste capítulo, observa-se que por diferentes especialistas trabalham no desenvolvimento de sistemas que auxiliam no processo de ensino e aprendizagem. Vários são os formatos de aplicação do computador na educação, assim como, também, são diversas as classificações dos sistemas utilizados. Entretanto, o grande desafio tem sido criar sistemas tutores capazes de

reagir às ações do estudante e, instantaneamente, tomarem decisões inteligentes, que propiciem um melhor aprendizado.

Abordagens mais recentes também têm sido utilizadas considerando estilos de aprendizagem, incluindo Algoritmos Genéticos no processo de adaptação do tutor ao perfil do aluno. A seção 2.9 faz referência a esse modelo.

2.9 Abordagens utilizando Algoritmos Genéticos

Dorça et al (2012) apresentaram uma comparação entre a aplicação das cadeias de Markov e Algoritmos Genéticos na construção de um sistema adaptativo ao perfil do estudante. Este sistema atua na detecção e no ajuste de estilos de aprendizagem. Os resultados mostraram que quando comparados às cadeias de Markov os Algoritmos Genéticos foram capazes de detectar os estilos de aprendizagem de forma mais rápida e fornecerem conteúdos personalizados, tornando o aprendizado mais fácil.

Para que um sistema seja considerado inteligente é necessário aplicar técnicas de Inteligências Artificial. O próximo capítulo apresenta a fundamentação teórica de IA, com ênfase nas Redes Neurais SOM utilizadas neste trabalho.

CAPÍTULO 3: REDES NEURAIS ARTIFICIAIS

A mente que se abre a uma nova ideia jamais voltará ao seu tamanho original. (Albert Einstein)

Não existe um conceito único para definir Inteligência Artificial (IA), porém, independente do autor ou da linha de IA, as definições mais conhecidas variam ao longo de duas dimensões. A primeira delas considera os processos de pensamento e o raciocínio, que medem os sucessos considerando a fidelidade ao desempenho humano. A segunda refere-se ao comportamento e tem seu sucesso medido em função do conceito de racionalidade - forma ideal de inteligência (RUSSEL E NORVIG, 2004).

Independente da definição de IA, diferentes áreas e disciplinas contribuíram para se chegar ao estado da arte: Filosofia, Matemática, Economia, Neurociência, Psicologia, Engenharia de Computadores, Teoria de Controle, Cibernética e Linguística, dentre outras.

Devido à dimensão do assunto e ao objetivo deste trabalho, é apresentado neste capítulo um breve histórico da parte da Inteligência Artificial denominada conexionista, prosseguindo com maior ênfase em um modelo particular de rede neural denominado Mapas de Kohonen ou ainda Mapas Auto-Organizáveis.

3.1 Sistemas Conexionistas

Também denominados Redes Neurais Artificiais (RNAs), os Sistemas Conexionistas são técnicas computacionais que propõem uma forma particular de processamento da informação com base na organização física do cérebro humano. Esses sistemas caracterizam-se pela reunião de uma grande quantidade de unidades processadoras (células ou neurônios) interligadas por um número expressivo de conexões que processam as informações de forma paralela.

O estudo de Sistema Conexionista não é recente e remonta à época da Gestação da Inteligência Artificial, fase que durou entre 1943 e 1955. O primeiro trabalho reconhecido como IA foi realizado na década de 1943 (MACCULLOCH E PITTS, 1943 apud CARVALHO et al, 1998), ocasião em que eles propuseram um modelo de neurônios artificiais.

Para propor o modelo de neurônios artificiais, McCulloch e Pitts se basearam no conhecimento da fisiologia básica e da função dos neurônios do cérebro, na análise formal da lógica proposicional criada por Russell e Whitehead e na teoria da computação de Turing (RUSSELL E NORVIG, 2004).

O neurônio humano é a unidade básica do cérebro e, isoladamente, pode ser considerado análogo a uma unidade de processamento. Ele combina estímulos de entradas e os processa resultando em uma saída.

A base do neurônio artificial é o neurônio humano, que nada mais é do que um pequeno módulo que simula o funcionamento deste último. A Figura 3.1 mostra o formato de um neurônio biológico e na Figura 3.2 é apresentado o formato

de um neurônio artificial.

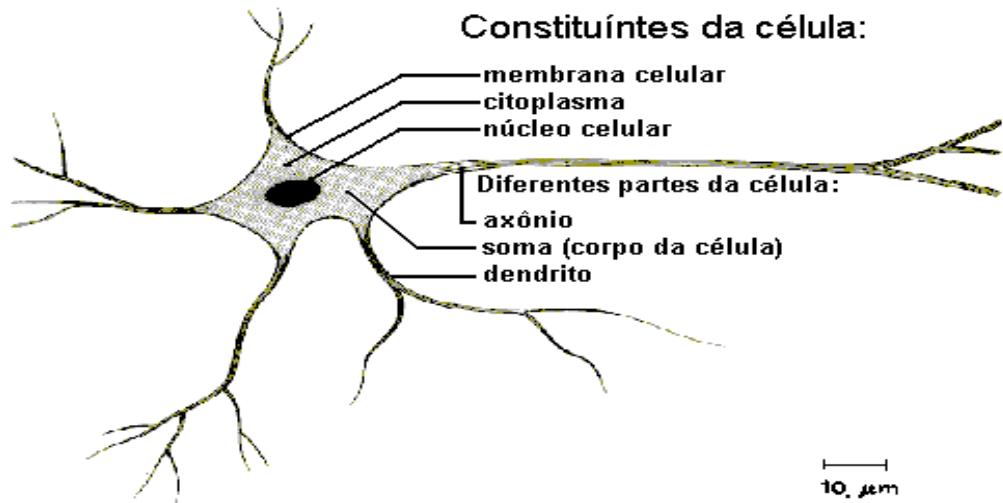


Figura 3.1 : Estrutura do neurônio humano (CARVALHO et al, 1998).

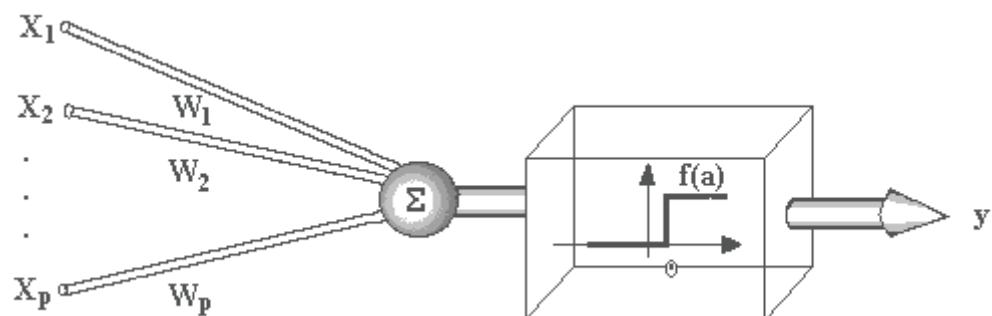


Figura 3.2: Estrutura do neurônio de MacCulloch e Pitts (CARVALHO et al, 1998).

O trabalho de McCulloch e Pitts (1943) concentrou-se, principalmente, em descrever um modelo artificial de um neurônio e em apresentar suas capacidades computacionais. As técnicas de aprendizado tornaram-se objeto de estudo apenas alguns anos depois, com as pesquisas apresentadas por Donald Hebb (1949 apud CARVALHO et al, 1998). Hebb mostrou como a plasticidade da aprendizagem de redes neurais é conseguida por meio da variação dos pesos de entrada dos

nodos (neurônios). Ele criou uma regra simples de atualização das intensidades de conexões entre os neurônios e também propôs uma teoria para explicar o aprendizado em nodos biológicos.

Em 1958, Rosenblatt (1958) apresentou o modelo de rede neural Perceptron e propôs um algoritmo para treiná-lo. Sua pesquisa demonstrou que o treinamento tornou o Perceptron capaz de executar determinados tipos de funções e classificar certos tipos de padrões.

Apesar da grande euforia ocorrida no início da utilização das redes neurais, em 1969 Minsky e Papert (1969 apud CARVALHO et al, 1998) provaram que esse modelo só era capaz de resolver problemas linearmente separáveis (problemas cuja solução pode ser alcançada dividindo-se o espaço de entrada em duas regiões, por meio de uma reta). Isto causou um esquecimento e uma consequente descrença nos sistemas baseados em redes neurais.

Paralelo a isso, estudos em Inteligência Artificial Simbólica continuaram sendo desenvolvidos e utilizados nas mais variadas aplicações. No paradigma simbólico, procura-se simular o comportamento inteligente utilizando-se técnicas baseadas em simbologias lógicas, entre outras, sem se preocupar com suas causas responsáveis.

Pesquisadores interessados em IA Conexionista também continuaram a desenvolver trabalhos nesse paradigma, ainda que timidamente. Entretanto, somente em 1986 Rumelhart, Hilton e Williams (1986) apresentaram o algoritmo back-propagation, que permite o treinamento de redes multicamadas. Eles demons-

traram que as redes neurais eram capazes de resolver, além de problemas linearmente separáveis, problemas com níveis muito maiores de complexidade. A partir de então, ocorreu uma explosão no interesse pelas redes neurais artificiais, também impulsionada pelos avanços da microeletrônica. Aliado a isso, o fato da escola simbolista (inteligência artificial simbólica) não ter conseguido grandes avanços na solução de determinados tipos de problemas também contribuiu para o ressurgimento das redes neurais artificiais (RUSSELL E NORVIG, 2004).

Atualmente existem diversos modelos de Redes Neurais Artificiais, com características distintas e aplicações nas mais diversas áreas. Alguns modelos adaptam-se melhor na solução de problemas de mineração de dados, outros no reconhecimento de voz, face etc. Independente da rede neural, o paradigma conexionista consolidou-se e atualmente é utilizado pelo governo (Receita Federal, Banco Central etc.), pela indústria e pelo comércio, entre outros.

Este trabalho utiliza o modelo proposto por Kohonen (1997) pela sua especificidade de se auto-organizar, permitindo agrupamentos de características e perfis. A seguir, é realizada uma abordagem desse modelo.

3.2 Mapas Auto-Organizáveis

Quando Minsky e Papert (1969 apud CARVALHO et al, 1998) demonstraram que o Perceptron só era capaz de resolver problemas linearmente separáveis, ocorreu um substancial desinteresse da comunidade acadêmica por pesquisas na área conexionista. Ainda assim, alguns entusiastas continuaram seus trabalhos nessa linha da IA. Na mesma década em que Rumelhart (1986) apresentou o algo-

ritmo back-propagation, o pesquisador Kohonen (1982) propôs um novo modelo de rede neural que recebeu as denominações de *Self Organizing Maps* (SOM), Mapas de Kohonen ou, em português, Mapas Auto-Organizáveis. A rede SOM teve sua concepção baseada no mapa topológico presente no córtex cerebral, como ilustrado na Figura 3.3. Esse modelo, como indica sua denominação, possui a capacidade de se auto organizar, preservando topologicamente a estrutura dos padrões apresentados.

Kohonen (1982) propôs um modelo em que seu algoritmo de treinamento utiliza o método de aprendizagem por competição. Na rede SOM, os neurônios próximos (na disposição do mapa) devem responder por funções similares (específicas), tal como no cérebro de animais mais evoluídos. Nesses casos, determinadas áreas respondem por estímulos de mesma natureza: fala, audição, visão etc.

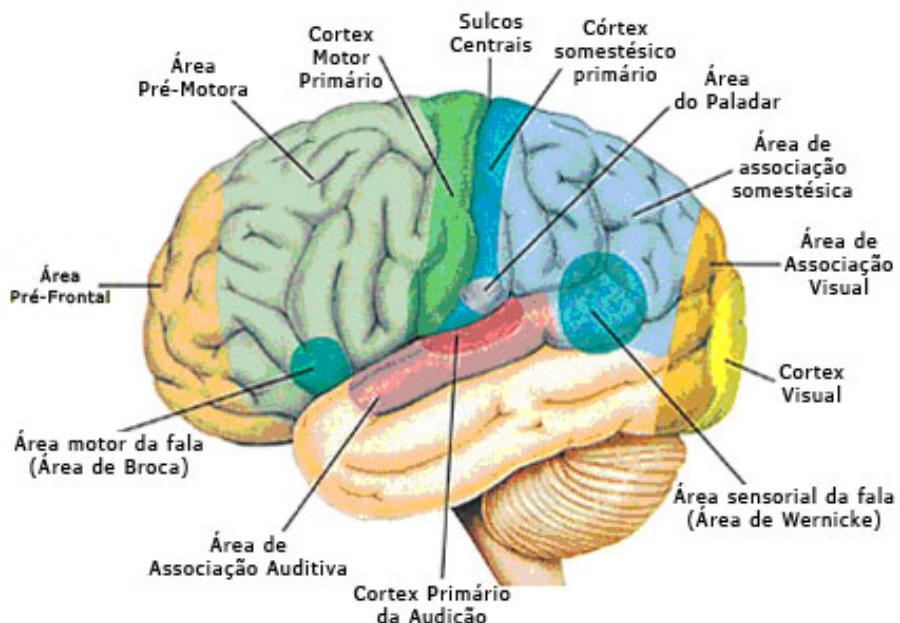


Figura 3.3: Córtex cerebral humano (SEELEY et al, 1996).

Os mapas de Kohonen são constituídos basicamente por uma camada de saída, que é a própria rede, além da camada de entrada que também pode ser entendida como uma camada sensorial (onde cada sinal do mundo externo é percebido por um determinado neurônio sensorial). A Figura 3.4 mostra a estrutura básica de uma rede SOM.

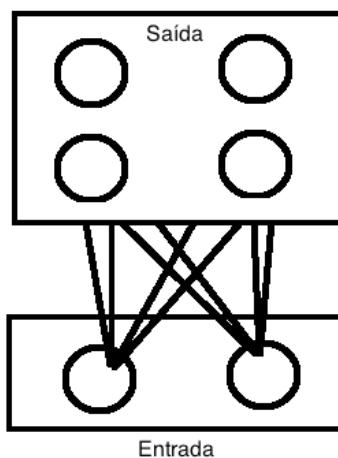


Figura 3.4: Modelo do mapa de Kohonen ou rede SOM.

Na rede SOM, os neurônios da camada de saída competem entre si para ver qual deles possui maior semelhança com o padrão apresentado. Entretanto, não existem conexões físicas entre eles.

Para melhor entendimento, é apresentado a seguir um breve estudo sobre a estrutura da rede SOM.

3.2.1 Estrutura da rede SOM

Os neurônios da camada de saída de uma rede SOM podem ser estruturados de diversas formas, denominadas de topologias. Uma topologia pode ser unidimensional, bidimensional ou, em alguns poucos casos, multidimensional. Po-

rém, na maioria das vezes ela é apresentada na forma de uma superfície plana (bidimensional).

Na rede SOM, todos os neurônios são localmente interconectados por uma relação de vizinhança que determina a topologia do mapa. O raio de vizinhança de uma rede varia de acordo com o treinamento, assumindo, após o treinamento, nenhuma vizinhança. Nesse caso, cada neurônio da rede já treinada deverá responder por um conjunto de entradas do mundo real, representando-o como saída da função de ativação.

A vizinhança entre os neurônios pode ser implementada de diversas maneiras, tais como linear em anel, hexagonal e retangular. As Figuras 3.5 e 3.6 (a) e (b) mostram os formatos de vizinhança em anel, hexagonal e retangular, respectivamente.

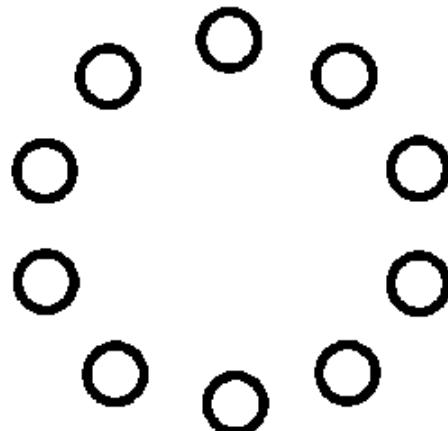


Figura 3.5: Formato para a região de vizinhança de uma SOM linear em anel.

Cada neurônio de um mapa de Kohonen é associado a um vetor de pesos ajustáveis “n-dimensional”, onde “n” corresponde ao número de entradas (característica) que cada padrão (exemplo do mundo real) possui. Durante o treinamen-

to, cada nodo da camada de entrada é apresentado a todos os neurônios da camada de saída, que disputam entre si para verificar qual deles poderá representá-lo (neurônio vencedor).

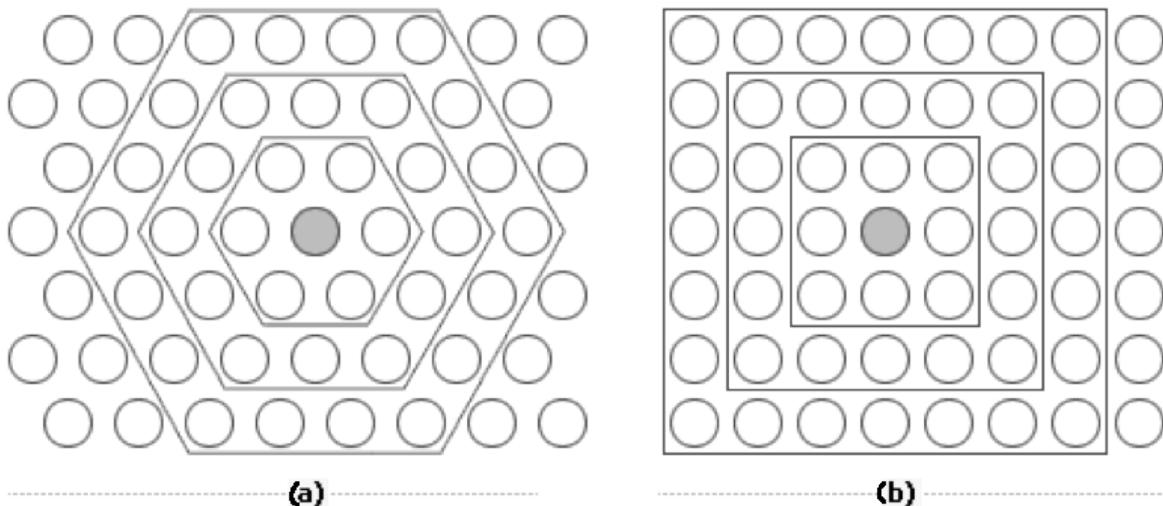


Figura 3.6: Vizinhança de uma rede SOM. a) hexagonal, b) regular.

Uma vez discutido sobre a estrutura dos mapas de Kohonen, mostra-se a seguir como é o processo de treinamento desse modelo.

3.2.2 Treinamento

Quando comparado ao treinamento de outros modelos de redes neurais mais clássicos, como ao do Perceptron Multicamadas (MLP, pelo inglês *Multi Layer Perceptron*), o treinamento da rede SOM tende a ser mais simples e rápido. Seu treinamento é realizado em duas fases distintas, denominadas ordenação global e ajuste fino (convergência), que serão descritas mais adiante neste capítulo.

Antes, porém, é realizada uma abordagem do conceito de ciclo, que serve de base para a compreensão do processo de treinamento.

3.2.2.1 Ciclos

Nos mapas de Kohonen, um ciclo é determinado por uma única apresentação à rede de todos os padrões a serem treinados. Se existem, por exemplo, cinquenta padrões compostos por três entradas cada um (sexo, idade e renda) que devem ser mapeadas por uma rede de dez neurônios, um ciclo é compreendido por uma única apresentação ao mapa dos cinquenta padrões (KOHONEN, 2001).

Como será visto adiante neste capítulo, a ordenação topológica da rede SOM ocorre nos ciclos iniciais, porém, por se tratar de um processo estocástico, a precisão do mapa depende do número de ciclos durante a fase de convergência ou ajuste fino. Não existe uma regra específica para a escolha do número de iterações (ciclos), mas estudos mostraram que quinhentos ciclos por neurônio são suficientes para se obter uma boa precisão estatística (KOHONEN, 1997). Portanto, a quantidade de ciclos pode ser estimada em função da quantidade de neurônios da rede.

Uma vez discutido o conceito de ciclo, retoma-se agora a discussão sobre o processo de treinamento da rede SOM, iniciando com um estudo sobre a fase de ordenação topológica.

3.2.2.2 Ordenação global

A fase de ordenação global de uma rede SOM, ou organização topológica do mapa, ocorre durante os mil primeiros ciclos (KOHONEN, 1997). Nessa fase o raio de vizinhança pode possuir um tamanho maior do que a metade do diâmetro da rede e diminui linearmente para uma unidade. A taxa de aprendizagem

também deve ser alta (próximo a 1) e reduzida gradualmente, em função do tempo, para um valor pequeno (próximo a 0,1 por exemplo).

Terminada a fase de ordenação global, a rede entra na etapa de convergência, objeto de estudo do próximo item deste capítulo.

3.2.2.3 Ajuste fino ou convergência

O ajuste fino ou fase de convergência é a fase mais demorada do treinamento e deve durar entre cem e mil ciclos a mais do que a fase de ordenação global, dependendo do tamanho da rede (quantidade de neurônios). O ajuste fino determina a qualidade de precisão do mapa.

Na convergência de uma rede SOM, a taxa de aprendizagem deve ser pequena, próximo a zero (0,01, por exemplo). Também é ideal que ela decresça em função do tempo. Neste estágio, a vizinhança também deve ser pequena: é aceitável uma unidade ou nenhuma vizinhança.

A Figura 3.7 mostra as disposições de um mapa com topologia bidimensional, durante as fases do treinamento. Pode-se verificar nessa figura que já nos primeiros mil ciclos de treinamento a rede possui a sua estrutura topológica razoavelmente definida.

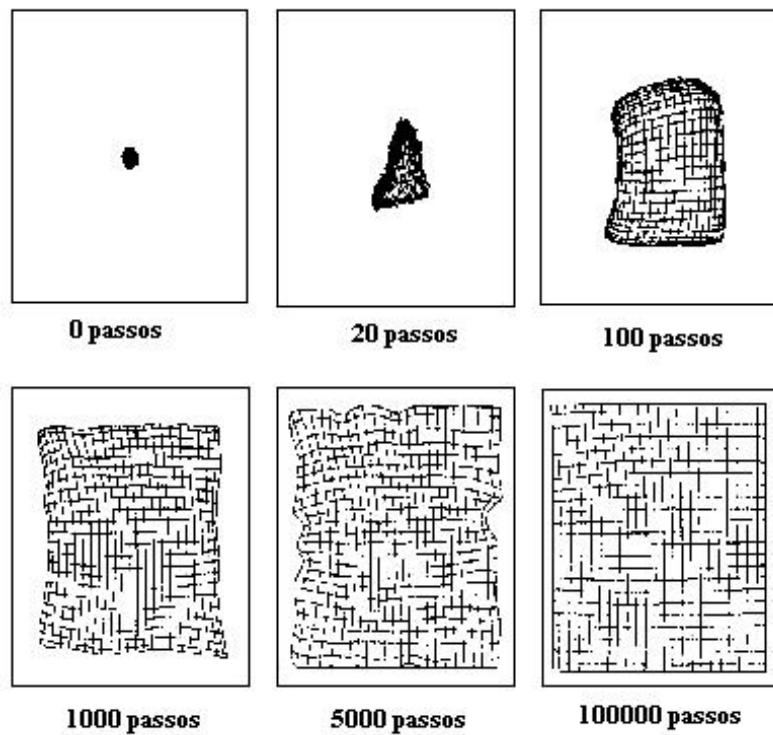


Figura 3.7: Fases do treinamento de uma rede SOM (KOHONEN, 1997)

Estudadas as fases de treinamento de uma rede SOM, é realizada a seguir uma abordagem sobre o algoritmo de treinamento.

3.2.3 Algoritmo de treinamento

Como já citado neste capítulo o mapa de Kohonen é uma rede neural em que o seu treinamento é rápido, quando comparado ao de outras RNAs. O algoritmo de treinamento desse modelo não é dotado de alto nível de complexidade, como pode ser observado na Figura 3.8. Variações desse algoritmo podem ser implementadas para adaptá-lo a aplicações específicas. Por exemplo, é possível criar rótulos associados aos neurônios. De modo simples, pode-se afirmar que os rótulos são vetores de probabilidades de decisão. Essa última modificação é utilizada neste trabalho e apresentada mais adiante no Capítulo 4 desta tese, que trata do modelo

proposto.

1. Inicializar pesos e taxa de aprendizagem
2. *Repetir*
 - 2.1. *Para cada* exemplo
 - 2.1.1. Apresentar o exemplo a todos os neurônios da rede e definir o neurônio vencedor (utilizando a distância euclidiana, por exemplo);
 - 2.1.2. Atualizar pesos do neurônio vencedor e de seus vizinhos (no raio de vizinhança estabelecido);
 - 2.1.3. *Se* satisfeita a regra de redução da taxa de aprendizado e vizinhança *então* reduzir a taxa de aprendizado e o raio de vizinhança;
 3. *Até* satisfeito o número de ciclos determinado ou característica do mapa não mudar.

Figura 3.8: Algoritmo básico de treinamento de uma rede SOM

Para melhor fundamentar o estudo sobre o processo de treinamento é exposto adiante como se determina um neurônio vencedor, citado no algoritmo mostrado na Figura 3.8.

3.2.4 Determinação do neurônio vencedor

Em uma rede SOM a determinação do neurônio vencedor pode ser realizada utilizando qualquer métrica. Porém, o mais usual é verificar qual neurônio possui a menor distância euclidiana em relação ao padrão apresentado (SILVA, 2000). Em um mapa bidimensional essa distância pode ser calculada pela Equação (3.1).

$$d = \left[\sum_j^n (x_{ji} - w_{ji})^2 \right]^{1/2} \quad (3.1)$$

onde:

d - distância euclidiana;

x_{ji} - i-ézimo padrão apresentado à rede;

w_{ji} - pesos sinápticos do neurônio i.

Definido o neurônio vencedor, o passo seguinte é fazer a atualização dos pesos. O processo de atualização de pesos é mostrado no item 3.2.5 deste capítulo, a seguir.

3.2.5 Ajuste de pesos

Na rede SOM os pesos são associados a cada neurônio e correspondem à quantidade de entradas de cada padrão. Por exemplo, se forem consideradas três entradas de cores R, G e B (*red, green e blue*), existe um vetor de pesos igual a três, como pode ser observado na Figura 3.9 e na Figura 3.10. O vetor que apresenta W_R , W_G e W_B , são os pesos associados às entradas R, G e B, respectivamente.

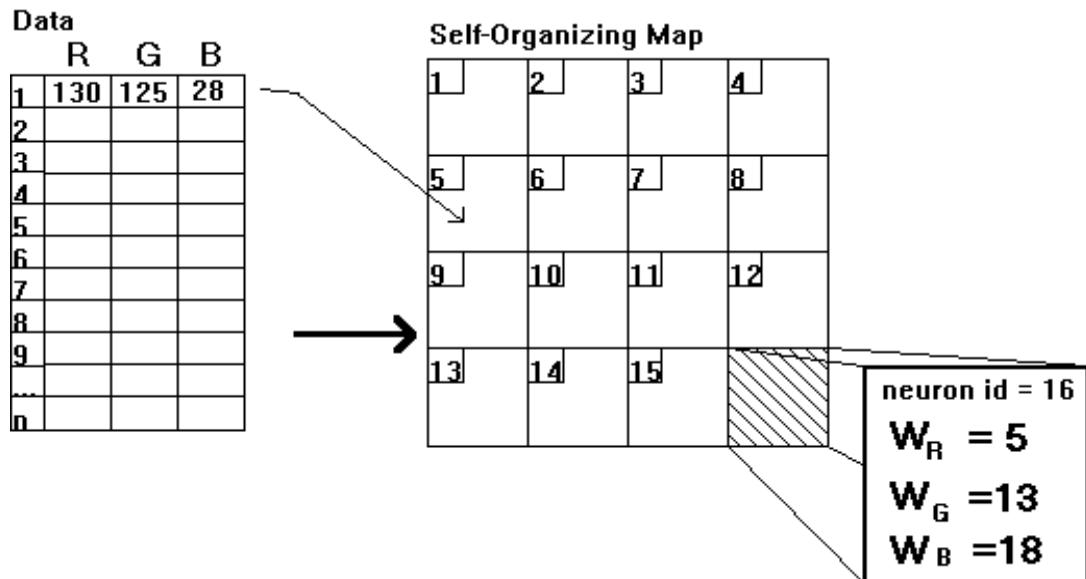


Figura 3.9: Matriz com pesos associados aos neurônios de uma rede SOM.

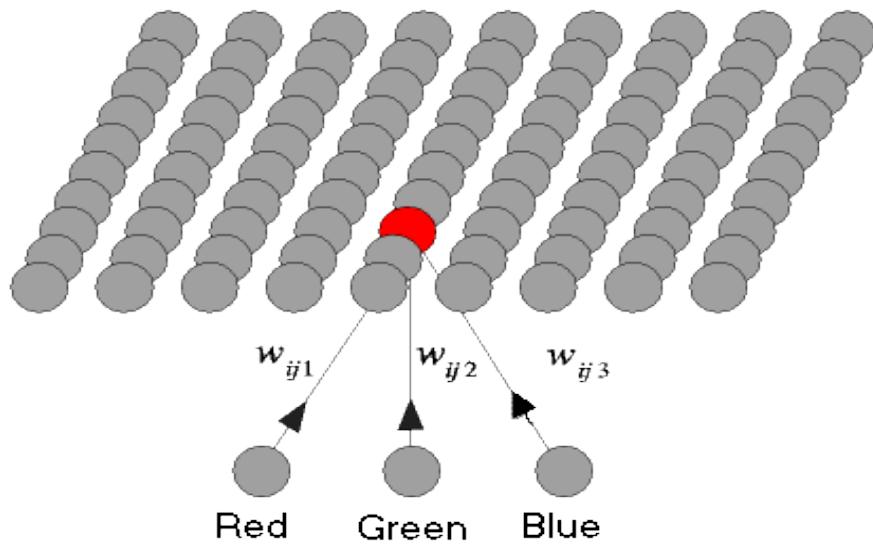


Figura 3.10: Visualização de pesos associados aos neurônios de uma rede SOM.

Em um mapa de Kohonen, a inicialização dos pesos pode ser realizada aleatoriamente com valores próximos a zero, dependendo da aplicação. Essa inicialização é mostrada mais detalhadamente no item 3.2.6 deste capítulo.

A atualização dos pesos em uma rede SOM bidimensional é realizada

como mostrado na Equação (3.2).

$$w_{ji}(t+1) = \begin{cases} w_{ji}(t) + \eta(t)(x_{ji}(t) - w_{ji}(t)), & \text{se } j \in \Delta(t) \\ w_{ji}(t), & \text{caso contrário} \end{cases} \quad (3.2)$$

onde:

$w_{ji}(t+1)$ - novo peso;

$w_{ji}(t)$ - peso atual;

η - taxa de aprendizado; e

x_{ji} - entrada do i -ézimo padrão apresentado à rede.

O método de atualização de pesos mostrado na Equação (3.2) deve ser aplicado ao neurônio vencedor e a todos os demais neurônios que estejam no seu raio de vizinhança.

No entanto, por se tratar de uma rede com aprendizado competitivo, muitas vezes é interessante fortalecer um raio de vizinhança mais próximo, garantindo que os neurônios imediatamente próximos sejam mais semelhantes. Ao mesmo tempo, também espera-se que se diferencie um raio de vizinhança intermediário, retornando ao ajuste padrão para o restante da rede. Esse tipo de função de atualização de pesos é denominado “chapéu mexicano”.

Com a função de atualização chapéu mexicano é possível atingir uma organização topológica mais eficiente. Neste caso são utilizados três níveis distintos de atualização de pesos dos vizinhos (taxas de aprendizagens diferenciadas). Assim, de acordo com o raio de vizinhança pode-se ter estímulos excitatórios, inibitórios e

levemente excitatórios. Nesse conceito de atualização, os neurônios mais próximos (dentro de um raio R_1), têm uma influência maior no seu grau de excitação; neurônios um pouco mais distantes (dentro de um raio R_2), têm um estímulo inibitório e os neurônios fora dessas áreas (dentro de um raio R_3) um estímulo levemente excitatório. A Figura 3.11 ilustra a forma do “chapéu mexicano”, utilizada na atualização de pesos de uma rede SOM.

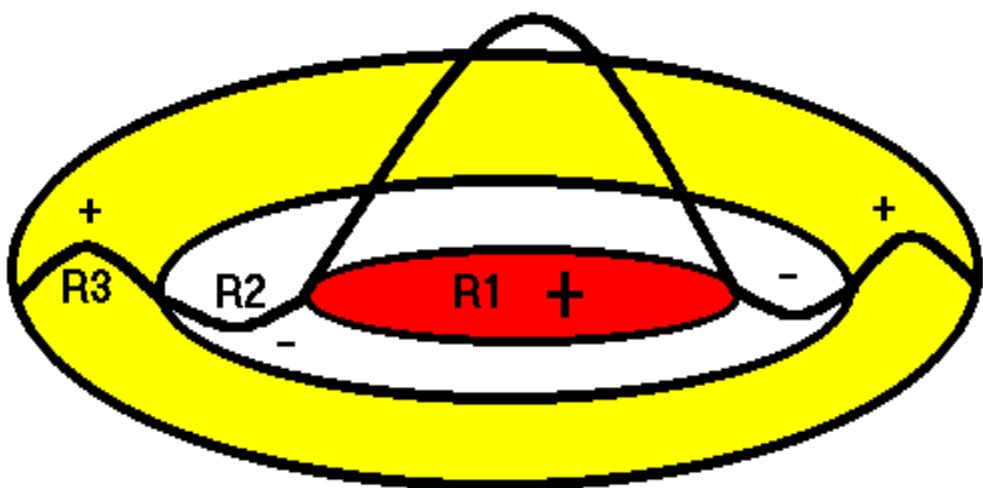


Figura 3.11: Chapéu mexicano utilizado na atualização de pesos de uma rede SOM.

Até este momento foram apresentados os preceitos básicos para o treinamento de uma rede SOM, sem questionar sobre os possíveis resultados apresentados por essa rede. No item 3.2.6 a seguir é estudado o processo de construção de mapas mais bem elaborados.

3.2.6 Como construir bons mapas

Apesar de ser possível construir o mapa de Kohonen sem muita precaução, existem algumas orientações que podem auxiliar no melhoramento da qua-

lidade desse mapa e no desempenho do treinamento. Pode-se citar que o treinamento de uma rede SOM é afetado pela inicialização dos pesos, taxa de redução da vizinhança, formato da vizinhança e pela taxa de aprendizado (SILVA, 2000). Para melhor fundamentar esse assunto apresenta-se no subitem 3.2.6.1 deste capítulo algumas observações que podem ajudar na construção de bons mapas.

3.2.6.1 Inicialização dos pesos

Habitualmente a inicialização dos pesos em uma rede SOM é realizada utilizando-se valores aleatórios. Todavia, nem sempre isso produz o melhor resultado. O motivo é que alguns vetores de pesos podem assumir valores muito diferentes dos padrões de treinamento, o que pode tornar alguns nodos da rede inutilizáveis. Neste caso, eles nunca poderão representar qualquer padrão do conjunto de exemplos. Uma alternativa para resolver esse problema é inicializar os pesos com amostras extraídas aleatoriamente do conjunto de exemplos. Outra opção é inicializar todos os pesos com valores nulos (zero).

Não menos relevante que a inicialização dos pesos é a forma com que a vizinhança é definida. Portanto, agora é necessário fazer uma abordagem sobre esse assunto.

3.2.6.2 Vizinhança

A escolha do melhor formato de vizinhança para um mapa de Kohonen pode ser empírica e depende de cada caso. Porém, para uma inspeção visual é preferível o formato hexagonal. Nesse modelo, não existe favorecimento a nenhum

dos sentidos horizontal ou vertical, presentes no modelo retangular. Um formato circular também pode não ser ideal, pois dificulta na estabilização e na orientação do mapa.

A redução do raio de vizinhança durante o treinamento é fundamental. Ela pode ser realizada de forma linear, desde que observados os estágios de ordenação global e de ajuste fino. Assim, já nos primeiros mil passos, o raio de vizinhança deve assumir um valor pequeno de uma ou nenhuma vizinhança.

Aplicando-se os métodos discutidos até aqui neste capítulo, é possível concluir o treinamento de uma rede SOM e obter um mapa cuja distribuição de grupos, ou ordenação topológica, seja realizada de forma estocástica.

Além das características de vizinhanças já mencionadas neste item, algumas aplicações exigem que os dados sejam mapeados para os locais previamente estabelecidos no mapa. A subseção 3.2.6.3 deste capítulo explica como implementar esse recurso.

3.2.6.3 Forçando a apresentação para os locais específicos do mapa

Para fazer com que uma região específica do mapa responda por padrões previamente determinados pode-se copiar os exemplos desejados para os vetores de pesos do ponto específico. Neste caso, torna-se necessário manter uma taxa de aprendizado baixa para essas regiões durante as fases de treinamento. Um modelo como esse se aplica à tarefa de agrupamento e nos casos em que é desejado manter determinadas características em um ponto específico do mapa, facilitando a vi-

sualização e o processo de decisão.

Até aqui foram mostrados os conceitos de padrões e suas entradas, sem aprofundar na forma de sua representação para que a rede possa compreendê-los. O subseção 3.2.6.4 deste capítulo faz uma abordagem sobre como tratar os padrões e sobre as formas de normalização desses.

3.2.6.4 Normalização de padrões

Uma dúvida comum a iniciantes na construção de modelos neurais é o não entendimento de como converter campos de textos, ou com grandes faixas de variações, para um formato padrão que possa ser compreendido pela rede.

O desejável é que todas as entradas tenham a mesma relevância. Portanto, todos os padrões de entradas deverão estar normalizados em uma faixa de valores, por exemplo, entre zero e um. Isso evita que uma dimensão se sobreponha a outra, tornando o aprendizado ineficiente.

O processo de normalização pode ser realizado convertendo os dados nominais para assumir qualquer valor dentro da faixa adotada. Por exemplo, o sexo masculino pode assumir o valor zero e o feminino o valor um. Para dados numéricos é possível aplicar um método que se baseie nos valores máximos e mínimos, equalizando para faixas entre zero e um. A Equação (3.3) mostra uma possível aplicação de normalização desse tipo.

$$x_{Normalizado} = \left(\frac{x - \min(x)}{\max(x) - \min(x)} \right) \quad (3.3)$$

Existem várias outras formas de normalizar os padrões e, em casos específicos, é possível que algumas entradas tenham maior relevância na tomada de decisão. Essas também poderão assumir faixas ligeiramente diferentes.

Aplicando-se as orientações mostradas neste capítulo tem-se um treinamento mais eficiente, mas não com o resultado mensurado. Para avaliar os mapas criados, podem ser aplicadas métricas de qualidade, estudadas na subseção 3.2.7 deste capítulo.

3.2.7 Métricas de qualidade

Por se tratar de um processo estocástico, nem sempre é fácil mensurar a qualidade do aprendizado da rede. Portanto, após o treinamento de uma rede SOM é fundamental saber o quanto o mapa se adaptou aos padrões apresentados. Para isso, torna-se necessário conhecer alguma métrica que permita medir a qualidade da rede. As formas mais usuais são verificar a precisão e a preservação topológica do mapa, mostradas nas subseções 3.2.7.1 e 3.2.7.2 deste capítulo.

3.2.7.1 Preservação topológica

Uma das características básicas de um mapa de Kohonen é a preservação topológica das características do conjunto de treinamento. Conhecer a qualidade dessa preservação é uma garantia inicial de que a rede aprendeu.

Uma forma simples de medir a qualidade topológica de uma rede SOM é a média da soma das distâncias entre o primeiro e o segundo neurônios vencedores. Assim, quanto maior o resultado obtido, menor é a ordenação global. A

Equação (3.4) mostra essa função para um mapa bidimensional:

$$QT = \frac{1}{N} \sum_{i=1}^n \alpha \quad (3.4)$$

onde:

QT – qualidade topológica

$$\alpha = \begin{cases} \max (|Lv - Lv'|, |Cv - Cv'|), & \text{se } \max (|Lv - Lv'|, |Cv - Cv'|) > 1 \\ 0, & \text{caso contrário} \end{cases}$$

Lv - linha do neurônio vencedor;

Lv' - linha do segundo vencedor;

Cv - coluna do neurônio vencedor;

Cv' - coluna do segundo vencedor; e

N - número de exemplos apresentados à rede.

A qualidade topológica avalia apenas o conceito de ordenação do mapa, mas não a precisão de resposta da rede. Sendo assim, uma métrica que avalie a precisão é desejável. Ela é apresentada na subseção 3.2.7.2 deste capítulo.

3.2.7.2 Precisão do mapa

Ao contrário da métrica de preservação topológica, a métrica de precisão de um mapa de Kohonen não se preocupa com a ordenação global. Procura-se nessa métrica, verificar com que precisão os neurônios respondem ao conjunto de exemplos treinados.

Se os vetores de pesos de uma rede SOM fossem exatamente iguais aos padrões de entradas, as distâncias euclidianas entre eles seriam nulas. Todavia, o que ocorre é a utilização de um número pequeno de neurônios para responder por uma malha grande de dados.

Uma forma de mensurar a precisão de um mapa de Kohonen é obter a média aritmética da soma das distâncias euclidianas de cada exemplo utilizado como base para o treinamento. Assim, quanto menor é o resultado (próximo a zero), melhor é a precisão. A Equação (3.5) pode ser aplicada, tomando como base a rede já treinada e os dados utilizados para o treinamento da rede.

$$\text{Precisão} = \frac{1}{N} \sum_{i=1}^N \| X_i - W_{ji} \| \quad (3.5)$$

onde:

x_i - i-ézimo padrão apresentado à rede; e

w_{ji} - peso do neurônio vencedor (referente à entrada).

Estudados o processo de treinamento de uma rede SOM, as orientações para a construção de bons mapas e os métodos para medir a qualidade, é importante exemplificar os passos iniciais de um treinamento, mostrados na subseção 3.2.8 deste capítulo.

3.2.8 Exemplo de uma rede neural SOM

Para concluir é apresentado um exemplo para mostrar os passos de treinamento de uma rede SOM composta por nove neurônios (mapa bidimensional 3

x 3). Neste exemplo são utilizados quinze padrões no treinamento, cada um deles compostos de três entradas R, G e B (*Red, Green e Blue*). Note que, para este caso, as entradas já estão normalizadas entre 0 (zero) e 255 (duzentos e cinquenta e cinco). O quadro da Tabela 3.1 mostra os padrões a serem treinados. Para o exemplo a taxa de aprendizado inicial assumida foi de $\eta = 0,99$ e o raio de vizinhança inicial $V=1$.

Tabela 3.1: Quadro de padrões a serem treinados no exemplo

Nº Exemplo	Red	Green	Blue	Cor resultante
1	255	255	255	Branca
2	0	0	0	Preta
3	255	0	0	Vermelha
4	255	255	0	Amarela
5	0	0	255	Azul
6	0	255	0	Verde
7	128	128	128	Cinza
8	255	128	0	Alaranjada
9	128	128	0	Abacate
10	255	0	255	Lilás
11	0	64	0	Verde escura
12	128	255	255	Azul piscina
13	128	64	0	Marrom
14	128	0	128	Roxa
15	0	0	128	Azul escuro

Para esse exemplo, os pesos iniciais da rede neural foram definidos aleatoriamente com valores pequenos. O quadro da Tabela 3.2 ilustra a disposição

inicial dos pesos da rede.

Tabela 3.2: Quadro com vetor de pesos iniciais do exemplo

$W_R=1$ $W_G=3$ $W_B=2$	$W_R=5$ $W_G=8$ $W_B=2$	$W_R=15$ $W_G=5$ $W_B=12$
$W_R=10$ $W_G=15$ $W_B=20$	$W_R=15$ $W_G=8$ $W_B=3$	$W_R=1$ $W_G=1$ $W_B=1$
$W_R=0$ $W_G=1$ $W_B=0$	$W_R=60$ $W_G=3$ $W_B=20$	$W_R=1$ $W_G=8$ $W_B=3$

A seguir são apresentados os passos iniciais para o treinamento da rede utilizada no exemplo.

1º Passo: Apresentar um padrão à rede e definir o neurônio vencedor.

Abaixo é detalhada a apresentação do primeiro padrão à rede.

$$\text{Neurônio (1, 1): Distância} = \sqrt{(255 - 1)^2 + (255 - 3)^2 + (255 - 2)^2}$$

$$\text{Distância} = 438,2111$$

$$\text{Neurônio (1, 2): Distância} = \sqrt{(255 - 5)^2 + (255 - 8)^2 + (255 - 2)^2}$$

$$\text{Distância} = 433,033486$$

$$\text{Neurônio (1, 3): Distância} = \sqrt{(255 - 15)^2 + (255 - 5)^2 + (255 - 12)^2}$$

$$\text{Distância} = 423,2599674$$

$$\text{Neurônio (2, 1): Distância} = \sqrt{(255 - 10)^2 + (255 - 15)^2 + (255 - 20)^2}$$

$$\text{Distância} = 415,7523301$$

$$\text{Neurônio (2, 2): Distância} = \sqrt{(255 - 15)^2 + (255 - 8)^2 + (255 - 3)^2}$$

$$\text{Distância} = 426,7469976$$

$$\text{Neurônio (2, 3): Distância} = \sqrt{(255 - 1)^2 + (255 - 1)^2 + (255 - 1)^2}$$

$$\text{Distância} = 439,9409051$$

$$\text{Neurônio (3, 1): Distância} = \sqrt{(255 - 0)^2 + (255 - 1)^2 + (255 - 0)^2}$$

$$\text{Distância} = 441,0963614$$

$$\text{Neurônio (3, 2): Distância} = \sqrt{(255 - 60)^2 + (255 - 3)^2 + (255 - 20)^2}$$

Distância = 395,9217094 (menor distância, vencedor)

$$\text{Neurônio (3, 3): Distância} = \sqrt{(255 - 1)^2 + (255 - 8)^2 + (255 - 3)^2}$$

$$\text{Distância} = 434,7746543$$

A menor distância euclidiana encontrada para o primeiro padrão apresentado à rede, foi em relação ao neurônio da linha três e coluna dois (em destaque). O passo seguinte é a atualização dos pesos do vencedor e de seus vizinhos, no raio de vizinhança definido.

2º Passo: Atualizar os pesos do neurônio vencedor, conforme abaixo.

$$W_{(3,2)R} = 60 + 0,99 * (255 - 60) = 253,05$$

$$W_{(3,2)G} = 3 + 0,99 * (255 - 3) = 252,48$$

$$W_{(3,2)B} = 20 + 0,99 * (255 - 20) = 252,65$$

Depois de atualizados os pesos do neurônio vencedor deve-se fazer o mesmo para cada neurônio vizinho no raio de vizinhança definido para o ciclo. Neste caso, a vizinhança adotada é igual a um.

Após as devidas atualizações, o vetor de pesos assume a forma mostrada no quadro da Tabela 3.3.

Tabela 3.3: Quadro do vetor de pesos após a primeira atualização

$W_R=1$ $W_G=3$ $W_B=2$	$W_R=5$ $W_G=8$ $W_B=2$	$W_R=15$ $W_G=5$ $W_B=2$
$W_R=252,55$ $W_G=252,6$ $W_B=252,65$	$W_R=252,6$ $W_G=252,53$ $W_B=252,48$	$W_R=252,46$ $W_G=252,46$ $W_B=252,46$
$W_R=252,45$ $W_G=252,46$ $W_B=252,45$	$W_R=253,05$ $W_G=252,48$ $W_B=252,65$	$W_R=252,46$ $W_G=252,53$ $W_B=252,48$

3º Passo: Se for satisfeita a regra de redução da vizinhança, então decrementar o raio de vizinhança (neste exemplo, satisfeitos os mil primeiros ciclos a vizinhança passa a ser nula).

4º Passo: Se for satisfeita a regra de redução da taxa de aprendizado, então decrementar “ η ” (neste exemplo, pode ser assumida uma redução linear simples do tipo $\eta(\text{nova}) = \eta(\text{antiga}) * 0,999$ para cada ciclo).

5º Passo: Volte ao primeiro passo e apresente o próximo padrão.

Após a apresentação de todo o conjunto de exemplos à rede e realizados os devidos ajustes de pesos, inicia-se um novo ciclo. Deve-se então reiterar o

processo, submetendo novamente os padrões à rede, até completar o número de ciclos determinado. Neste caso, quinhentos ciclos por neurônio são suficientes, ou seja, quatro mil e quinhentos ciclos.

O exemplo apresentado teve a intenção de mostrar, de forma simples, o início do treinamento de uma rede SOM. Neste caso não existe a pretensão de aprofundar em detalhes, como a redução da taxa de vizinhança, a redução da taxa de aprendizagem, entre outros.

O próximo capítulo apresenta a proposta de um sistema tutor inteligente híbrido dotado de características adaptativas baseadas nas decisões do professor, subsidiadas pelas redes neurais SOM.

CAPÍTULO 4: MODELO PROPOSTO

O planejamento não é uma tentativa de predizer o que vai acontecer. O planejamento é um instrumento para raciocinar agora, sobre que trabalhos e ações serão necessários hoje, para merecermos um futuro. O produto final do planejamento não é a informação: é sempre o trabalho. (Peter Drucker)

O desenvolvimento de Sistemas Tutores Inteligentes é desafiador e exige conhecimentos multi e interdisciplinar, com abrangência nas áreas de Engenharia de Computação, Educação, entre outras. A elaboração da proposta de um modelo inclui ainda a definição da abrangência de atuação do sistema, além de especificar o nível de interferência do professor (especialista) no sistema e na sua forma de agir.

O sistema proposto possui uma estrutura híbrida em que as decisões são realizadas, inicialmente, utilizando os conhecimentos do professor responsável pela elaboração do conteúdo teórico. Essa estrutura de navegação sugerida pelo professor servirá de orientação para o estudo do aluno e deve ser validada e, quando necessário, substituída pelas decisões de uma rede neural SOM. O modelo de rede neural escolhido pode ser encontrado nas mais diversas aplicações, devido à sua capacidade de ordenação topológica e de classificação de perfis ou padrões (CARVALHO et al, 1998).

A estratégia de ensino inicial do sistema tutor híbrido (inteligente) é modelada pelo professor. Porém, as decisões sugeridas e construídas pelo professor, serão, gradualmente, refinadas pela rede neural.

Os padrões de treinamento do modelo neural responsável pelo refinamento das regras do professor são extraídos da navegação dos estudantes, no tutor com decisão especialista. A consequente adaptação do tutor é também baseada na forma com que os estudantes se comportam no sistema, ou seja, navegam pelos conteúdos do curso. Isto atribui características de maior proximidade da forma humana de agir. No modelo proposto a base de domínio passa a ter suas decisões considerando, além do conhecimento do professor, o conjunto das características mapeadas pelas redes neurais artificiais, que passam a ser as responsáveis por parte das tomadas de decisões.

Os testes realizados neste trabalho utilizaram um curso em que o conteúdo versa sobre conhecimentos básicos de informática. Naturalmente, o curso foi organizado de acordo com as estruturas dos tutores, a serem apresentadas ainda neste capítulo. A seguir é realizada uma breve descrição do ambiente tecnológico utilizado neste trabalho.

4.1 Ambiente Tecnológico do Sistema

A Internet por meio da *World Wide Web* (WWW) tem sido amplamente utilizada nos mais diversos seguimentos e com os mais variados objetivos, incluindo a área comercial (negócios online), troca de informações científicas entre os pesquisadores, institutos de pesquisas, empresas de ciência e tecnologia etc. A

WWW também está promovendo uma revolução na área educacional, pois oferece benefícios como intercâmbio entre alunos e professores, facilidades a pesquisas, atendimento *online* (bate-papo ou *chat* para tirar dúvidas), *fórum* de discussão, redes sociais, entre outros. No intuito de usufruir de todas essas vantagens, neste trabalho optou-se pelo ambiente web para utilização do software tutor.

Para esta tese foram desenvolvidos dois programas. O primeiro deles é um tutor com decisão especialista, utilizado inicialmente como ambiente de estudo e, ao mesmo tempo, como recurso para a coleta de dados necessários ao treinamento das redes neurais. O segundo programa consiste da adaptação do primeiro modelo, acrescentando às regras do sistema as decisões da rede neural. Este modelo é um tutor inteligente híbrido, baseado nas redes neurais artificiais SOM. Ambos foram construídos com o uso da tecnologia Ruby ®, atuante no lado do servidor. Essa tecnologia proporciona dinamismo às páginas web e acesso à base de dados.

Na base de dados estão armazenadas todas as informações necessárias ao treinamento das redes neurais, bem como os dados inerentes à navegação do aluno, permitindo inclusive realizar levantamentos estatísticos. A obtenção dos valores das variáveis temporais e de comportamento (latências, satisfação etc.) foi realizada com a utilização da linguagem de script JavaScript. Isto garante maior precisão e fidedignidade aos registros, uma vez que essa linguagem atua no lado cliente (navegador) evitando assim os retardos inerentes à comunicação. O treinamento das redes neurais foi executado com a utilização da tecnologia Ruby.

A seguir é apresentada a estrutura básica adotada nos tutores com de-

cisão especialista utilizado na coleta de dados e híbrido, propostos neste trabalho.

4.2 Estrutura dos Tutores com Decisão Especialista e Híbrido

A primeira etapa deste trabalho consistiu na modelagem e desenvolvimento de uma estrutura que permitisse, além da coleta de dados com decisão especialista, a sua adaptação para o modelo com decisão neural. A utilização de uma estrutura similar, porém com tomada de decisão diferenciada, permite avaliar os resultados sem que uma mudança de ambiente seja uma variável que possa interferir neste ou naquele resultado.

Portanto, os tutores com decisão especialista e híbrida possuem estrutura e características similares, como a mostrado na Figura 4.1 inspirada na proposta de Carvalho (2002).

O modelo permite a um professor construir cursos que versem sobre variados temas e disciplinas. No entanto, deve-se fazer a adequação das unidades didáticas à estrutura básica do tutor. Além dos textos, o professor deve também elaborar os exercícios com questões objetivas, que são utilizados nas avaliações da retenção de conteúdo a curto prazo, ou, de forma simples, da aprendizagem (ALBUQUERQUE, 2001).

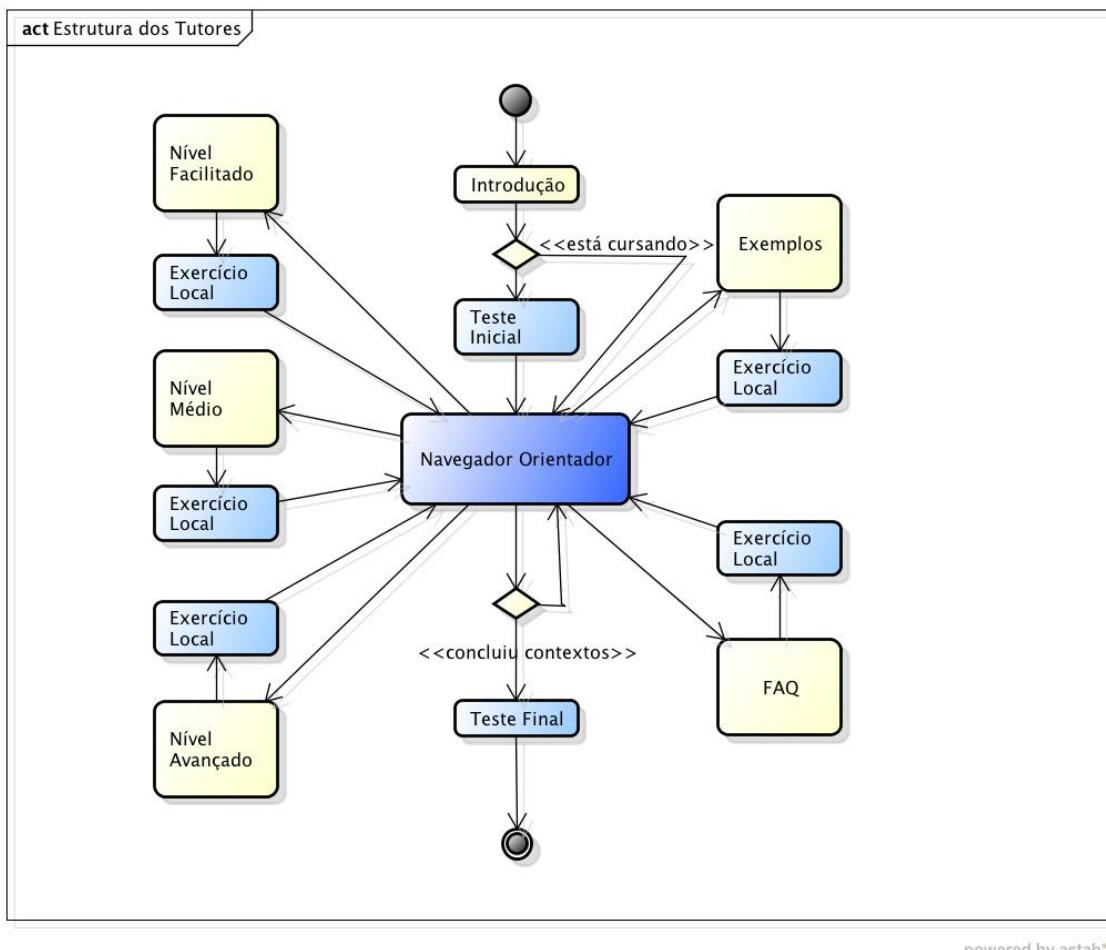


Figura 4.1: Estrutura dos tutores especialista e híbrido adaptada de Carvalho (2002).

Na estrutura proposta neste trabalho o curso a ser ofertado deve ter as suas unidades didáticas organizadas e distribuídas em níveis que tem como objetivo desenvolver diferentes competências. Portanto, os cursos são constituídos por uma sequência de conteúdos, dispostos em cinco níveis distintos, além dos testes inicial e final.

Desses cinco níveis, três podem ser considerados como principais: facilitado, médio e avançado; e os outros dois como níveis auxiliares: respostas a perguntas frequentes (FAQs) e exemplos. Cada um dos níveis deve ter uma abordagem textual adequada a seu propósito e independentemente do nível tem-se ainda a

opção de incluir uma figura representativa associada ao conteúdo estudado.

A distribuição em níveis teve sua inspiração na transposição didática de conteúdos. Neste conceito, a necessidade de ensinar leva à modificação do próprio conhecimento, convertendo-o em “objetos de ensino” passível de aprendizado por parte do estudante. Esse é um processo natural executado pelo professor durante sua atividade docente (CHEVALLARD, 1998). Sendo assim, o conteúdo do conhecimento pode ser selecionado ou recortado de acordo com o que o professor considera relevante para construir as competências esperadas.

A seguir, é realizado um detalhamento das características esperadas para cada um dos níveis disponibilizados na estrutura do tutor proposto nesta tese.

O teste inicial busca verificar o conhecimento que o estudante tem sobre o conteúdo a que será exposto. Esse teste avalia com base nos conteúdos de nível médio, pressupondo que se possa a partir de um perfil médio mensurar o domínio a respeito do assunto.

O nível facilitado consiste de textos com linguagem simples e direta. Esse nível faz abordagem ao conteúdo estudado naquele ponto, sem considerar suas particularidades. Ele permite ao aprendiz ter um conceito básico do assunto em questão.

O nível médio apresenta características mais abrangentes do que o nível facilitado, mas sem aprofundar em informações mais específicas do conteúdo em pauta.

O nível avançado conduz o estudante a uma abordagem mais aprofun-

dada, além da proposta pelo nível médio, oferecendo características técnicas de categoria avançada. A Figura 4.2 mostra a tela de textos do sistema proposto neste trabalho.

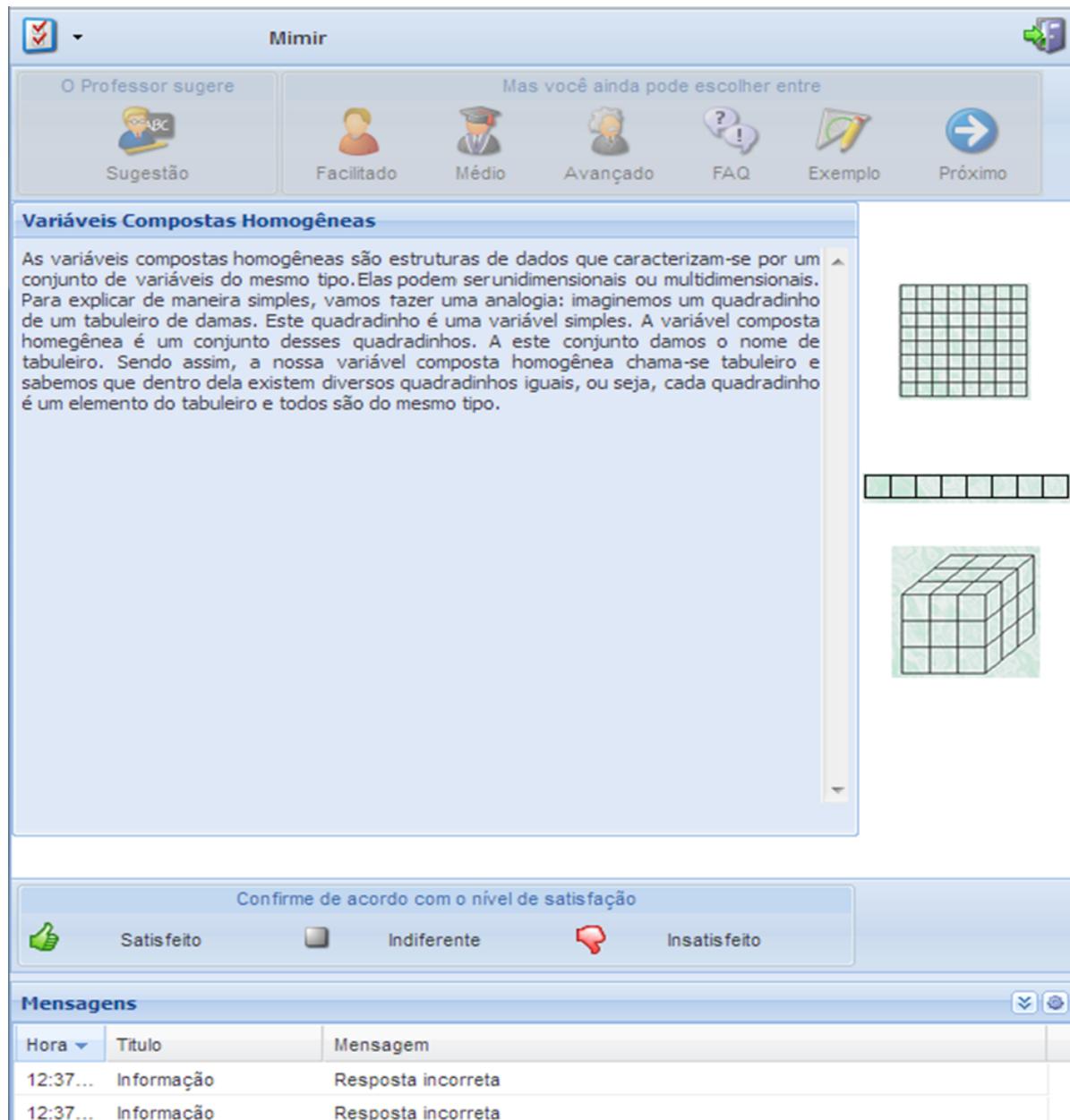


Figura 4.2: Tela de texto do tutor proposto neste trabalho.

As respostas às perguntas mais frequentes, denominadas neste trabalho simplesmente de FAQs, consistem de uma série de perguntas e respostas de ca-

ráter geral, que são normalmente apresentadas por pessoas que estudam aquele conteúdo.

Os exemplos são textos que possuem estrutura lógica com o objetivo de exemplificar o conteúdo apresentado. Neste nível, são utilizadas analogias, contextualizações etc.

O teste final é disposto no intuito de avaliar a aprendizagem do aluno durante a execução do tutor e tal como o teste inicial tem sua constituição com base no nível médio. Outra finalidade do teste final é a de permitir a validação dos dados, comparando se ocorreu ou não aprendizado por parte do estudante a partir da comparação entre os testes inicial e final.

Os testes inicial e final são constituídos basicamente de questões de múltipla escolha, contendo um enunciado e proposições, sendo uma delas a correta. Além desses, cada um dos níveis (facilitado, médio, avançado, FAQs e exemplos) é seguido de uma questão relacionada ao conteúdo daquele contexto e nível, também de múltipla escolha. A Figura 4.3 mostra a tela de exercícios do sistema.

The screenshot shows the Mimir tutor interface. At the top, there's a navigation bar with icons for Sugestão, Facilitado, Médio, Avançado, FAQ, Exemplo, and Próximo. Below the navigation bar, a question titled "Questão 1" is displayed:

Fazendo uma analogia: imaginemos um quadradinho de um tabuleiro de damas. Este quadradinho é uma variável simples. A variável composta homegênea é um conjunto desses quadradinhos. A este conjunto damos o nome de tabuleiro. A variável composta homogênea chama-se tabuleiro e sabemos que dentro dela existem diversos quadradinhos iguais, ou seja, cada quadradinho é um elemento do tabuleiro e todos são do mesmo tipo. Se os quadrados fossem diferentes, qual seria o nome da estrutura de dados?

To the right of the question, there is a 8x8 grid icon representing a chessboard.

Below the question, a "Respostas" section contains five radio button options:

- Variável do tipo literal
- Variável do tipo real
- Variável do tipo inteiro
- Variável composta Homogênea
- Variável composta Heterogênea

At the bottom of the interface, there is a satisfaction survey section with three buttons: Satisfeito (green thumbs up), Indiferente (grey square), and Insatisfeito (red thumbs down).

Finally, a "Mensagens" section shows a list of messages:

Hora	Título	Mensagem
12:34...	Informação	Escolha um curso!
12:34...	Bem vindo	Luiz Loja

Figura 4.3: Tela de exercícios do tutor proposto neste trabalho.

O processo de execução do tutor basicamente consiste na apresentação sequencial de cada um dos conteúdos do curso em seus diversos níveis. Dessa forma, depois de aplicado o teste inicial, o estudante é encaminhado ao início do curso

no nível mais adequado ao seu perfil, considerando o resultado obtido nesse teste. Após estudado esse nível e respondido o exercício correspondente o aluno é exposto a uma página de navegação, como mostrado na Figura 4.4.



Figura 4.4: Navegador do tutor proposto neste trabalho.

De acordo com o desempenho do aluno a página de navegação orienta sobre qual é o nível mais adequado de estudo, ou, até mesmo, indica que ele deve prosseguir para um novo assunto. No modelo com decisão especialista a orientação é realizada utilizando o conhecimento do professor e no modelo híbrido proposto pela combinação da decisão do professor e da RNA. Mesmo assim, o estudante ainda pode consultar outros níveis ou prosseguir para o próximo conteúdo de acordo com a sua necessidade e interesse. Mesmo após ter visitado outros níveis, se o estudante desejar prosseguir o tutor irá orientá-lo para o nível mais adequado considerando o seu desempenho médio na utilização do tutor. Depois de apresentados todos os conteúdos do curso, o aluno é encaminhado para o teste final que consiste de uma série de exercícios.

O estudante poderá visitar cada um dos níveis dos conteúdos uma única vez. Após essa visita o “*link*” é desabilitado automaticamente pelo sistema. Essa decisão está baseada no interesse em avaliar o aprendizado com base na transposição didática de conteúdos diferentes e não no reforço ou reapresentação de conteú-

dos já estudados.

As páginas de conteúdos, bem como os exercícios, contêm botões de confirmação que devem ser escolhidos de acordo com o nível de satisfação do usuário. Isto deve ser realizado obedecendo aos critérios apresentados no quadro da Tabela 4.1.

Tabela 4.1: Quadro de botões de resposta com níveis de satisfação.

Botão de opção	Descrição
 Satisffeito	Nível de satisfação excelente. O estudante gostou muito do conteúdo ou exercício.
 Indiferente	Nível de satisfação normal. O estudante não percebeu excepcionalidade no conteúdo ou exercício.
 Insatisffeito	Nível de satisfação ruim. O estudante não gostou do conteúdo ou exercício.

A estrutura apresentada até aqui é a base para a construção dos tutores com decisão especialista e modelo híbrido propostos neste trabalho. As seções 4.3 e 4.4 deste capítulo, mostram com mais detalhes esses modelos.

4.3 O Tutor com Decisão Especialista

Uma vez definida a estrutura básica a ser adotada pelos modelos, a etapa seguinte deste trabalho foi a construção de um tutor com decisão especialista em que seja possível sua utilização imediatamente após a inserção do conteúdo teórico. Esse modelo tem características adaptativas, de acordo com as definições apresentadas pelo professor, ao mesmo tempo em que sua utilização produz os dados necessários para o treinamento da rede neural adotada no sistema tutor híbrido.

A denominação tutor com decisão especialista é porque é função do professor indicar os critérios de navegação do estudante, ou seja, construir a estratégia de ensino. Estas decisões são tomadas levando-se em consideração o desempenho global, para a indicação do próximo conteúdo a ser estudado, e o desempenho local, para a orientação do estudo de outros níveis do mesmo assunto. As subseções 4.3.1 e 4.3.2 explicam em detalhadamente as formas de deslocamentos entre os conteúdos e os níveis, respectivamente.

4.3.1 Deslocamento do eixo entre os conteúdos

Nas propostas de Carvalho (2002) e Melo et al (2005), quando o aluno concluía o estudo de um determinado conteúdo ou mesmo decidisse prosseguir para um outro assunto, o sistema o encaminhava para o nível médio da próxima unidade didática. Assim, era exigido que o aluno estudasse obrigatoriamente o nível médio de todos os conteúdos de um curso.

No modelo proposto neste trabalho, o eixo de navegação que encamina o aluno para um próximo conteúdo a ser estudado foi deslocado do nível médio para um nível que melhor se adapte ao perfil do estudante. Isso é realizado considerando-se o desempenho global do aluno, indicado pelo professor quando ele cobra o curso. Assim, o professor pode decidir que estudantes que tenham um resultado abaixo ou acima de uma determinada média possam ser submetidos a níveis diferentes do médio na próxima unidade didática a ser estudada. Dessa forma, alunos com desempenho acima da média podem, por exemplo, visitar com mais frequência os níveis avançados. Por outro lado, alunos com desempenho inferior podem ficar

mais restritos a visitas a níveis facilitados. Com isso, espera-se que o sistema torne-se mais atrativo e motivador pois os estudantes poderão adaptar-se melhor ao estudo, uma vez que o próprio tutor irá reagir considerando o seu nível de conhecimento do tema em estudo. A Figura 4.5 apresenta a interface de cadastro de cursos no sistema proposto neste trabalho.

Figura 4.5: Interface de cadastro de cursos no sistema proposto neste trabalho.

O desempenho global é calculado pelo resultado da avaliação inicial, quando o curso estiver sendo iniciado, ou pelo percentual de exercícios respondidos corretamente em todos os conteúdos previamente estudados, quando o curso já estiver em andamento. Portanto, na tomada de decisão, o sistema calcula o desempenho global do aluno e o compara com o que foi estabelecido pelo professor, para então orientar o nível da próxima unidade didática.

O deslocamento entre os níveis de um mesmo conteúdo segue o critério de desempenho local apresentado na subseção 4.3.2 deste capítulo.

4.3.2 Deslocamento entre os níveis de um mesmo conteúdo

Quando o aluno inicia o estudo de um conteúdo ou assunto, ele é direcionado para o nível mais adequado ao seu desempenho global. Como visto anteriormente, o aluno que apresenta um resultado acima de determinada média, pode ser direcionado para o nível avançado do próximo conteúdo. Todavia, ao estudar esse conteúdo o aluno pode não apresentar um desempenho local compatível com o global. Nesse caso, é necessário fazer um ajuste local da decisão e sugerir ao aluno que estude um outro nível desse mesmo conteúdo. Assim, o aluno que não conseguir um bom desempenho no estudo do nível avançado, por exemplo, deve ser encaminhado localmente para um outro nível da mesma unidade didática. Para possibilitar a decisão local, o professor especialista tem à sua disposição durante a elaboração do curso a opção de indicar qual a melhor sugestão de estudo para o aprendiz, caso ele acerte ou erre o exercício local de uma determinada unidade didática. Por exemplo, o professor pode decidir que um determinado estudante que não acertou um exercício de nível médio seja submetido a um nível facilitado da mesma unidade didática. Por outro lado, em um conteúdo mais complexo, o professor pode considerar que é mais interessante para o estudante ser encaminhado para um conjunto de exemplos. A Figura 4.6 mostra a interface de cadastro de conteúdos pelo professor.

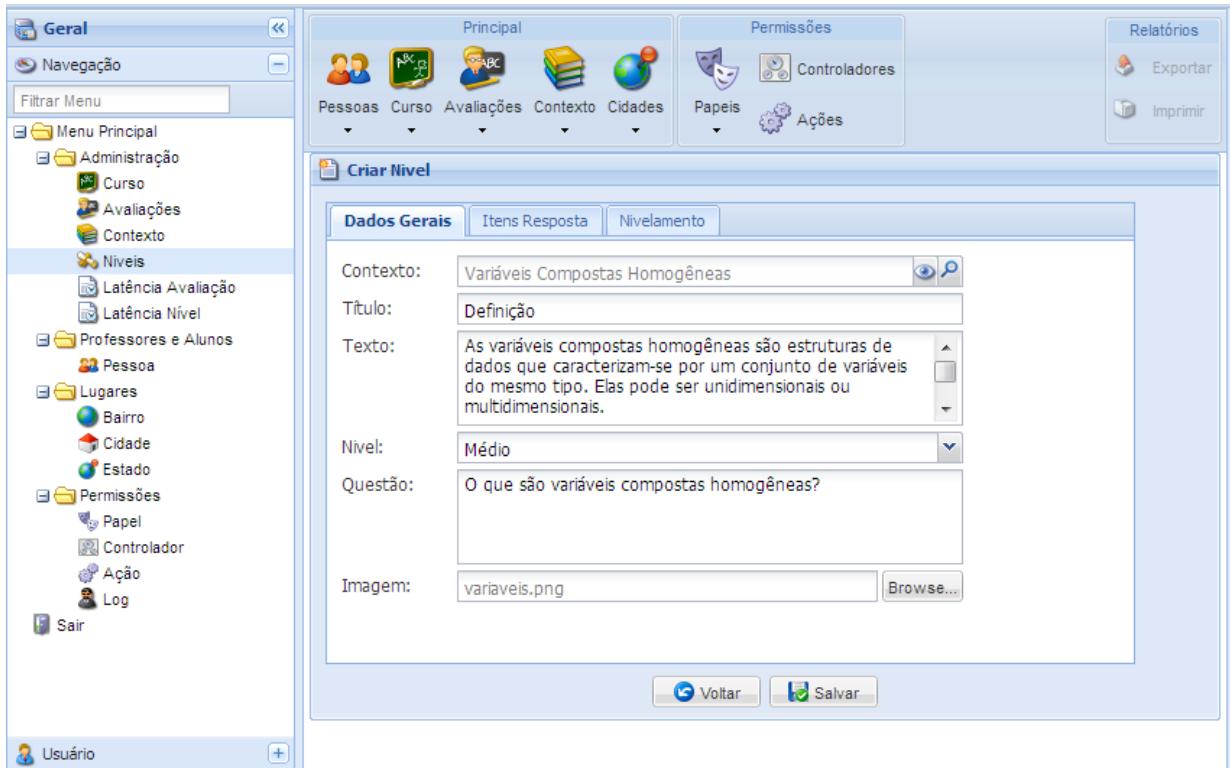


Figura 4.6: Tela de cadastro de conteúdos do modelo proposto neste trabalho.

Com a finalidade de facilitar o trabalho do professor na construção dos cursos, o sistema sugere os critérios de navegação local ou decisão especialista, com base na pesquisa apresentada por Melo et al (2005). Na ocasião, Melo aplicou um questionário para professores de diversas instituições de ensino, com o objetivo de mapear as melhores decisões a serem adotadas em cada situação de resposta do estudante a um determinado exercício. A pesquisa considerava apenas os cinco níveis utilizados na estrutura do sistema proposto neste trabalho, sem, no entanto, avaliar o conteúdo de um curso em particular. Os resultados dessa pesquisa são utilizados apenas como sugestão. O professor, construtor do curso, tem toda a liberdade de modificar as orientações especialistas de acordo com seus saberes disciplinares, experenciais e até mesmo da sua formação profissional, considerando a especificidade do tema abordado (TADIRF, 2002).

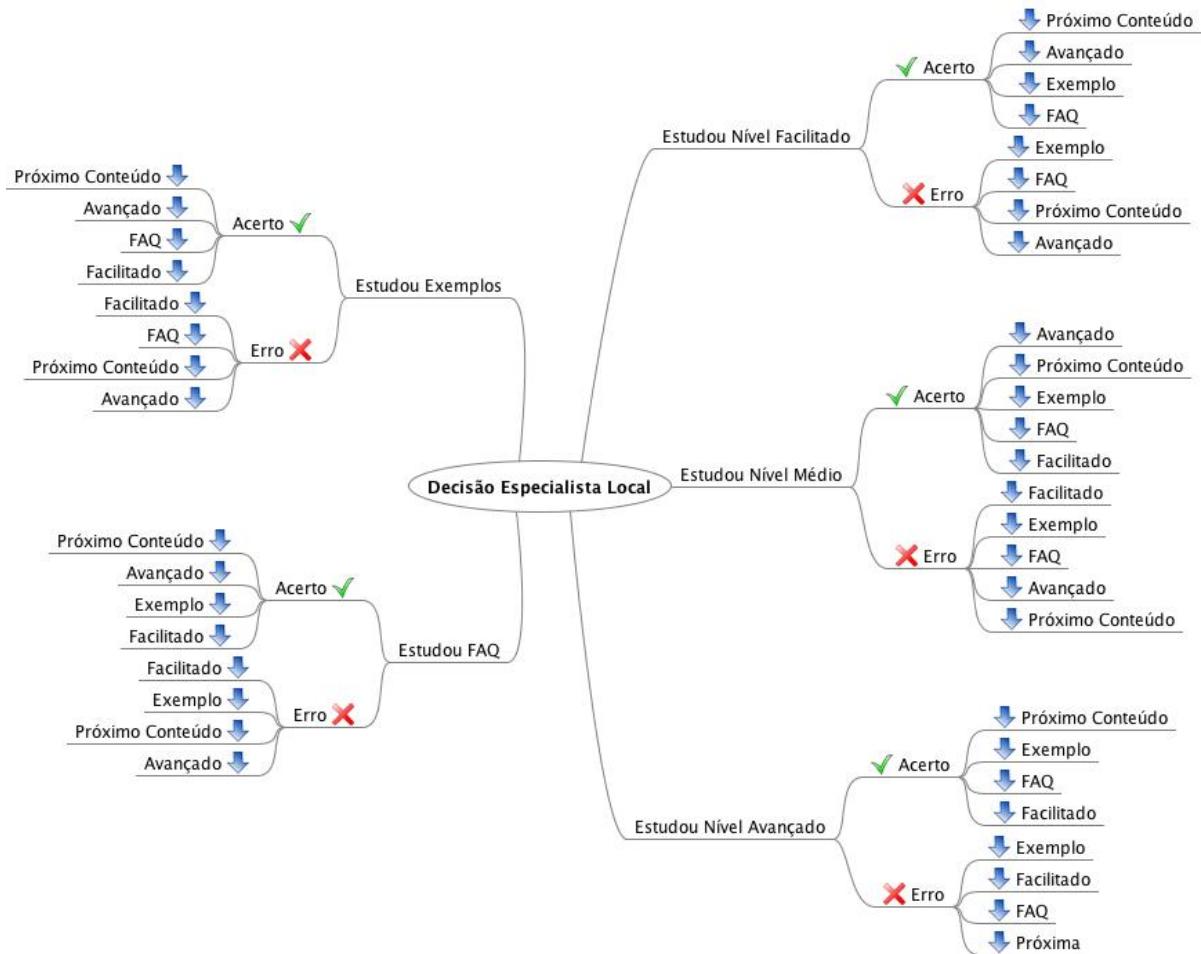


Figura 4.7: Árvore de sugestão de decisão local.

A Figura 4.7 apresenta um mapa simplificado das sugestões da decisão especialista segundo o nível de acerto ou erro de um exercício local. Portanto, caso o estudante responda de maneira incorreta a um exercício referente ao nível médio, por exemplo, os passos seguintes são classificados e existirá uma sequência de possíveis indicações.

O sistema tutor com decisão especialista apresentado neste item pode ser utilizado imediatamente após a criação do curso como instrumento de auxílio ao aprendizado. Ao mesmo tempo, ele atua como um mecanismo de coleta de dados para o treinamento das redes neurais utilizadas no modelo híbrido descrito a seguir.

4.4 O Tutor Híbrido

No tutor com decisão especialista mostrado na seção 4.3 deste capítulo, utilizado para coleta de dados, compete exclusivamente ao professor decidir qual é o passo seguinte a ser indicado para o estudante. A base dessa decisão é o conhecimento do especialista sobre o assunto, bem como a sua experiência docente. Todavia, sabe-se que a tarefa de ensino aprendizagem é complexa e a formação de um professor não é definida com precisão.

Por outro lado, pode-se também observar que o conceito de indivíduo é um fator que pode tornar a generalização pouco viável, especialmente no processo de estudo. Assim, aplicar regras gerais na tomada de decisão pode não ser abrangente o suficiente para atender às especificidades de características individuais ou de perfis de estudo de indivíduos.

Com a finalidade de minimizar as distorções da decisão especialista, sejam elas por descuido ou por falta de domínio do ambiente ou conteúdo, propõe-se neste trabalho a criação de um modelo híbrido. Este modelo utiliza as redes neurais para auxiliar na orientação do estudo. A subseção 4.4.1 apresenta a estrutura básica do modelo proposto nesta tese.

4.4.1 Estrutura do tutor com decisão híbrida

O tutor híbrido segue a mesma estrutura de distribuição de conteúdos do tutor com decisão especialista ilustrada na Figura 4.1, ou seja, testes inicial e final; níveis facilitado, médio, avançado, FAQs e exemplos. A diferença entre esses

modelos é quanto à forma de decisão utilizada na orientação da navegação. No tutor híbrido, as orientações apresentadas no “navegador”, também existentes no modelo especialista, são subsidiadas pelas decisões da rede neural artificial. É função da RNA auxiliar na tomada de decisão de qual é o próximo passo que o estudante deve seguir.

Assim, tanto no tutor híbrido quanto no tutor com decisão especialista, o estudante é submetido às avaliações inicial e final, e, também, às opções de navegação de todas as unidades didáticas. Em ambos os tutores, é apresentado ao estudante a orientação do sistema tutor, seja ela puramente especialista ou auxiliada pela decisão neural. Entretanto, o estudante tem a liberdade de escolher qual nível visitar e em qual ordem fazê-lo. A rede neural tem o papel de apoiar a decisão do professor, com base nos padrões de comportamento dos estudantes que se submeteram à navegação no sistema tutor com decisão especialista. Com isso, o sistema é capaz de orientar o educando para a navegação no nível que seja mais adequado ao seu perfil, incluindo uma nova unidade didática, quando for o caso.

As características dos textos e dos exercícios seguem inalteradas no tutor com decisão híbrida, ou seja, são semelhantes às do tutor com decisão especialista. Isto porque os dados coletados durante a navegação especialista, destinados ao treinamento das redes neurais, devem ser coerentes com os dados gerados pelo uso do tutor híbrido. Desta forma, mantém-se em ambos os tutores a mesma correlação de layouts, cores, textos etc.

No tutor híbrido, assim como no tutor especialista, cada passo do es-

tudante é registrado em um banco de dados. Os registros resultantes são utilizados para a submissão à rede neural, para a decisão do próximo passo, bem como para a comparação dos resultados entre os dois tutores. Esses dados também são utilizados para novos treinamentos da rede, de modo que o sistema tenha um treinamento automatizado e contínuo.

Portanto, de acordo com as respostas da RNA o tutor híbrido tem suas estratégias de ensino refinadas pelo “modelo do aluno”. O sistema faz uma comparação entre a decisão do professor e aquela apresentada pela rede neural. A decisão do especialista somente prevalece nos casos em que a rede incorre em um erro grave. São considerados erros graves as incoerências de resposta da RNA, tais como indicações para os níveis em que não ocorra concordância com o desempenho local ou global do estudante. Por exemplo, é considerado um erro grave a sugestão da rede de encaminhar um estudante para um nível avançado quando esse não tiver obtido sucesso em um nível facilitado.

Quando comparado às propostas de Melo et al (2005) e Martins e Carvalho (2004), em que é necessária a prévia coleta de dados e, consequentemente, o treinamento das redes neurais, o modelo proposto neste trabalho faz o treinamento de forma automatizada. Ele permite que um curso seja disponibilizado para estudo após a inserção do conteúdo e, com isso, dispensa o trabalho da coleta de dados em sistemas não reativos ao perfil do estudante ou não dotados de conhecimento prévio. Além disso, o tutor híbrido possui um treinamento automatizado para a rede neural, que é apresentado na subseção 4.4.3 deste capítulo. Esse recurso dispensa a interfe- rência de um especialista em redes neurais na definição de aspectos como a quanti-

dade de ciclos para o treinamento, o número de neurônios da rede, entre outros.

Portanto, busca-se com a estrutura apresentada neste trabalho, obter-se um tutor dinâmico, capaz de adaptar-se durante sua execução às características humanas de agir, construindo e reconstruindo a sua estratégia de ensino. O modelo permite ao STI híbrido, subsidiado pelas respostas das redes neurais, orientar o educando para os níveis de conteúdos mais adequados ao seu desempenho ou perfil de estudo.

Para o treinamento da RNA é necessário definir as entradas da rede neural que compõem o conjunto de treinamento e a base de consulta ao modelo. A subseção 4.4.2 aborda esse assunto com mais detalhes.

4.4.2 Entradas das redes neurais

Um fator complexo na criação de uma rede neural é definir o conjunto de entradas dos exemplos, para compor a base de treinamento. Neste estudo, são consideradas as características comuns na maioria das pessoas quando essas desenvolvem atividades de estudo. Outros trabalhos utilizaram respostas a exercícios, rolagem da páginas, tempo para responder os exercícios, entre outros (DORÇA et al, 2009).

Neste contexto, foram consideradas nove entradas para as redes neurais: latência de leitura, grau de satisfação, latência de resposta do exercício, grau de satisfação em relação ao exercício, desempenho local, desempenho global, resultado médio obtido no teste inicial, unidade didática e nível atuais. A seguir, cada uma

dessas entradas é explicada em maiores detalhes.

O tempo ou latência de leitura do texto é o tempo gasto pelo estudante, em milissegundos, para concluir a leitura do texto que é apresentado a ele. O grau de satisfação em relação ao texto é o conceito de aceitação do estudante, relativo ao texto apresentado, distribuído em três níveis (satisffeito, indiferente e insatisffeito). O tempo ou latência de resposta do exercício é o tempo gasto pelo estudante, em milissegundos, para responder ao exercício de uma determinada unidade didática. O grau de satisfação em relação ao exercício é similar ao grau de satisfação do texto, mas referente ao exercício. O desempenho local refere-se ao acerto ou erro do exercício de um nível de uma unidade didática. Por sua vez, o desempenho global é obtido pela razão entre o número de exercícios respondidos corretamente nas unidades didáticas estudadas e o total de questões respondidas. O resultado obtido no teste inicial é a razão entre o número de questões respondidas corretamente no teste inicial e o total de questões deste. Pressupondo que a decisão de próximo ponto a ser estudado está relacionada com o conteúdo atual, considerar-se-á, ainda, como entrada para as redes, a unidade didática e o nível estudados. Isso possibilita um agrupamento também por conteúdos de estudos e seus respectivos níveis.

4.4.3 O treinamento automatizado da rede

Independente do modelo adotado, o processo de treinamento de uma rede neural é uma tarefa complexa que exige experiência e conhecimento empírico de um engenheiro do conhecimento. Em particular nas redes SOM, alguns fatores interferem diretamente no desempenho do aprendizado. Entre eles, pode-se mencionar:

nar a definição da quantidade de neurônios, a forma de inicialização dos pesos, a taxa de redução da vizinhança, o formato da vizinhança e a taxa de aprendizado, entre outros (KOHONEN, 2001). Com isso, modificações em sistemas que utilizam essa tecnologia podem implicar em novos treinamentos das redes. Nesses casos é necessário avaliar a qualidade topológica e a precisão da nova rede e, se necessário, ajustar para um modelo que melhor represente o conjunto de treinamento.

No modelo proposto neste trabalho foi desenvolvido um algoritmo que faz o treinamento de modo automatizado. Dessa forma, minimiza-se a necessidade de se ter um profissional em redes neurais para avaliar a qualidade das redes e fazer ajustes a cada curso criado. Além disso, esse modelo possibilita que um novo curso possa ser disponibilizado para estudos imediatamente após a sua criação. À medida que o sistema é utilizado com o conhecimento especialista, produz-se o conjunto de treinamentos. Quando existirem dados suficientes, o sistema se adapta de forma automatizada e passa a ter suas decisões subsidiadas pelo modelo neural. O sistema pode ser configurado para treinar as redes quando existirem, por exemplo, um conjunto de seiscentos registros. Caso tenha um número muito grande de padrões na base de dados, o sistema também pode ser configurado para utilizar no treinamento somente dados dos estudantes que obtiveram melhor aprendizado no tutor especialista, limitando os padrões a cinco mil, por exemplo.

Para possibilitar o treinamento automatizado, a topologia e o formato da vizinhança da rede neural são previamente definidos. Já a redução do tamanho da vizinhança ocorre em função da quantidade de ciclos de treinamento e o número de ciclos depende da quantidade de neurônios, segundo orientações do criador do mo-

delo neural (KOHONEN, 2001).

A Figura 4.8 apresenta o algoritmo básico do treinamento automatizado.

1. Obter total de padrões válidos (padrões que promoveram acerto no próximo)
2. Obter a dimensão máxima de redes a treinar
por exemplo, raiz quadrada do (total de padrões válidos / 7)
3. Obter a dimensão mínima de redes a treinar
por exemplo, raiz quadrada do (total de padrões válidos / 20)
4. Obter o total de redes a treinar
por exemplo, (dimensão máxima subtraído da dimensão mínima)
5. Para cada uma das redes a treinar
 - 5.1. Calcular a dimensão da rede a treinar
por exemplo, somando a dimensão mínima com a rede em treinamento
 - 5.2. Calcular o total de ciclos da rede a treinar
por exemplo, quadrado da dimensão atual multiplicado por 500
 - 5.3. inicializar pesos e taxa de aprendizagem
 - 5.4. Repetir
 - 5.4.1. Para cada exemplo
 - 5.4.1.1. Apresentar o exemplo a todos os neurônios da rede e definir o neurônio vencedor (utilizando a distância euclidiana, por exemplo)
 - 5.4.1.2. Atualizar pesos do neurônio vencedor e de seus vizinhos (no raio de vizinhança estabelecido)
 - 5.4.1.3. Se satisfeita a regra de redução da taxa de aprendizado e vizinhança, então reduzir a taxa de aprendizado e o raio de vizinhança
 - 5.5. Até satisfeito o número de ciclos determinado ou característica do mapa não mudar
 6. Calcular qualidades topológica e de precisão
 7. Obter rede com melhor qualidade topológica e de precisão e associar ao curso

Figura 4.8 Algoritmo básico do treinamento automatizado

Para definir a rede ideal para cada curso, o algoritmo treina redes bi-

dimensionais de diferentes dimensões e seleciona a que apresenta o melhor resultado de qualidade topológica e de precisão. Buscando garantir a generalização no aprendizado, como mostrado na Figura 4.8 o algoritmo controla ainda o número de neurônios em função da quantidade de padrões de treinamento. Isso significa que o algoritmo de treinamento automatizado estima as quantidades mínima e máxima de neurônios, com base na quantidade de padrões disponíveis para o curso e, em seguida, define as dimensões das redes a serem treinadas. Esse processo deve ser repetido para cada novo curso inserido no sistema.

Como pode ser observado no algoritmo da Figura 4.8, depois que a rede é treinada, o algoritmo calcula as qualidades topológica e de precisão, para que a rede com melhor resultado possa ser utilizada pelo curso no sistema tutor híbrido.

O modelo de treinamento automatizado e a forma de implementação do próprio tutor possibilita que existam várias redes diferentes no sistema, uma para cada curso.

Depois de treinadas as redes e definida a dimensão que melhor se adaptou ao curso, faz-se a rotulação dos neurônios, conforme apresentado na subseção 4.4.4 deste capítulo.

4.4.4 Rotulação das redes neurais

A rede neural SOM tem apenas uma camada e na sua forma mais simples ela é utilizada apenas para informar qual é o neurônio vencedor ou o que mais se adaptou (ajustou suas sinapses convenientemente) a um determinado padrão. No

caso deste trabalho, cada neurônio deve ser capaz de decidir entre cinco níveis distintos de orientação (cinco classes pré-definidas). Assim, a rede neural além de determinar o neurônio vencedor, deve indicar qual dos próximos possíveis níveis o estudante deverá ser orientado a estudar. Por exemplo, a partir do nível médio o estudante pode visitar o nível facilitado, avançado, FAQs ou exemplos daquela mesma unidade didática ou prosseguir para um novo conteúdo.

Para permitir os neurônios tomarem decisões variadas, foi criado um processo de rotulação que consiste em associar cada neurônio ao nível que ele deve sugerir. A Figura 4.9 apresenta a ilustração do processo de rotulação.

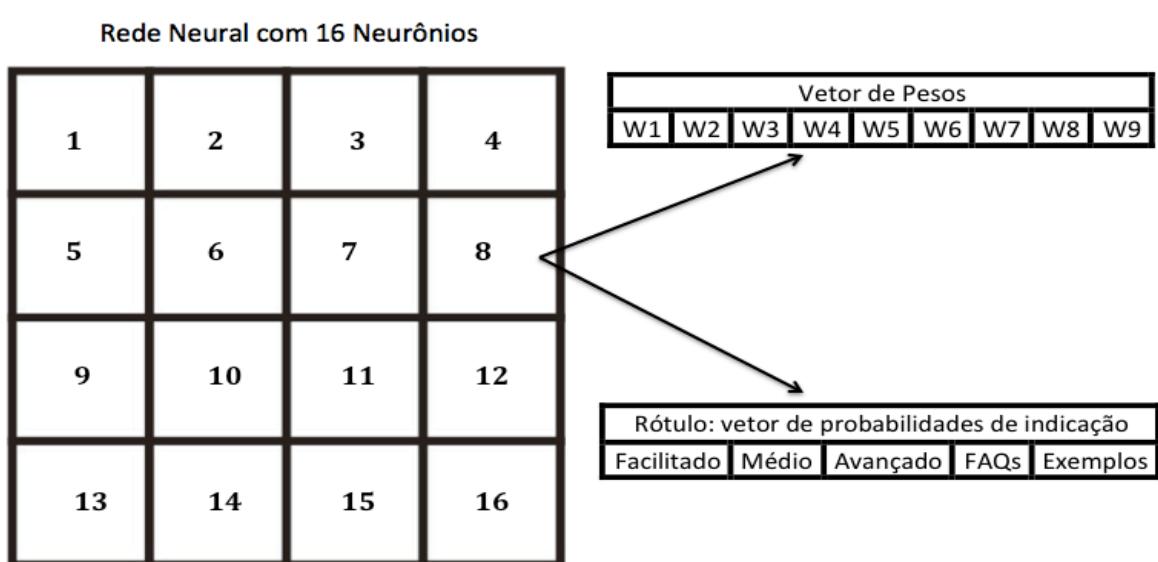


Figura 4.9: Ilustração do processo de rotulação dos neurônios

Como pode ser observado na Figura 4.9, cada um dos neurônios está associado a um vetor de pesos e a um outro vetor de probabilidade de indicação denominado rótulo. O rótulo é obtido a partir do conjunto de treinamentos. No exemplo, o neurônio oito está associado a um vetor de probabilidade indicando cada um dos níveis que ele deverá orientar o estudante, além do vetor de pesos equivalente às

nove entradas da rede neural (1 - latência de leitura, 2 - grau de satisfação, 3 - latência de resposta do exercício, 4 - grau de satisfação em relação ao exercício, 5 - desempenho local, 6 - desempenho global, 7 - resultado médio obtido no teste inicial, 8 – unidade didática e 9 - nível atual). Dessa forma, caso o nível a ser orientado já tenha sido visitado pelo estudante, a rede neural ainda terá a possibilidade de indicar um outro nível com menor probabilidade.

A rotulação ocorre apenas depois do treinamento das redes neurais, quando cada exemplo treinado é mapeado na rede, reforçando qual o próximo passo que o neurônio deverá indicar. Isto é feito com base na decisão seguinte do estudante (bem sucedido) que gerou o padrão.

O algoritmo mostrado na Figura 4.10 mostra o processo de rotulação utilizado neste trabalho.

1. Para cada Exemplo

1.1. Obtenha o passo seguinte do aluno que gerou o Exemplo

1.2. Para cada Neurônio da Rede

1.2.1. Incremente a rotulação do próximo passo que o neurônio deve indicar (de acordo com o passo seguinte efetuado pelo aluno que gerou o exemplo), utilizando a distância euclidiana, por exemplo

1.3. Até concluir a rotulação em todos os neurônios

Figura 4.10: Algoritmo de rotulação da rede SOM (para todos os neurônios).

Pode-se observar no algoritmo da Figura 4.10, que todos os neurônios

da rede têm atualizados o rótulo indicador de passo seguinte. Para permitir isto, foi associado a cada neurônio um vetor com os cinco possíveis níveis de indicação, como mostrado na Figura 4.9. O processo de rotulação é executado por meio da atualização, em cada neurônio, do nível (rótulo) equivalente à próxima visita efetuada pelo aluno gerador do padrão. Essa atualização resume-se em acrescentar, no ponto de indicação do neurônio (equivalente à próxima visita do padrão), o valor de semelhança (distância euclidiana) entre o padrão apresentado e os pesos do neurônio. Isto implica na rotulação de todos os neurônios da rede, todavia o neurônio sómente deverá indicar o nível onde mais se repetiram os padrões semelhantes (pontos fortalecidos).

Outra forma de criar a rotulação atualiza o próximo passo apenas do neurônio vencedor. Porém, estudos demonstraram que isso pode ocasionar alguns problemas com relação aos neurônios intermediários. Neurônios que se situavam em uma área de transição e que não representavam suficientemente bem nenhum exemplo específico do conjunto de treinamento algumas vezes se tornavam incapazes de tomar qualquer decisão. Isto acontece porque durante o processo de rotulação tais neurônios não eram ativados como vencedores nenhuma vez, o que impossibilitava a sua rotulação. Mesmo assim, esses neurônios deveriam ser capazes de responder por padrões que os ativassem durante a execução do sistema.

O algoritmo ilustrado na Figura 4.11 exemplifica a forma alternativa de rotulação mencionada acima.

1. Para cada Exemplo

- 1.1. Obtenha o passo seguinte do aluno que gerou o Exemplo
- 1.2. Obtenha o neurônio vencedor (que deverá representar o Exemplo)
- 1.3. Incremente a rotulação do próximo passo que o neurônio vencedor deve indicar (de acordo com o passo seguinte efetuado pelo aluno que gerou o exemplo).

2. Até concluir a rotulação em todos os exemplos

Figura 4.11: Algoritmo de rotulação da rede SOM (apenas neurônios vencedores).

Concluídas as etapas do sistema proposto, o próximo capítulo mostra os resultados obtidos nos testes realizados para esta tese.

CAPÍTULO 5: RESULTADOS OBTIDOS

O que sabemos é uma gota, o que ignoramos é um oceano. (Isaac Newton)

Para concluir o resultado da pesquisa, tem-se o desafio de avaliar o comportamento do modelo proposto e a aplicabilidade das tecnologias e dos métodos utilizados em sua construção. Portanto, este capítulo apresenta a análise dos experimentos e os resultados obtidos.

Inicialmente, é mostrada a metodologia incluindo o processo de coleta de dados e a consequente composição do conjunto de treinamento das redes neurais, realizados na primeira fase deste trabalho. Em seguida, são apresentadas as etapas do treinamento automatizado e as comparações do processo de estudo nos tutores.

5.1 Realização do Experimento

Neste experimento foi realizada a pesquisa de campo utilizando os modelos especialista e híbrido, para a coleta de dados e validação dos resultados.

Para a primeira coleta de dados foi construído um sistema tutor com a decisão orientada pelo especialista, de onde foram gerados os padrões para o treinamento da rede neural utilizada no tutor híbrido.

A segunda etapa da coleta foi realizada com o modelo híbrido, com o objetivo de avaliar o comportamento do modelo proposto e comparar os resultados.

A análise dos dados é realizada em dois momentos: utilizando os dados da primeira coleta no tutor com orientação especialista para a validação dos padrões usados no treinamento das redes neurais e utilizando os dados do modelo híbrido, para a comparação dos resultados entre os modelos.

Conhecidas as características do modelo proposto e a necessidade de dados para o treinamento das redes neurais, o experimento se desenvolve em etapas. Assim, a construção e a utilização do sistema com orientação especialista deve anteceder à do tutor híbrido, uma vez que o primeiro serve de base para o desenvolvimento do segundo.

5.1.1 Formatação do conteúdo

Construir um conteúdo adequado à estrutura proposta não é fácil e consiste no trabalho de um especialista da área, com habilidades também em tutoria. Portanto, é necessário, além do conhecimento técnico, ter o domínio da docência. Estas habilidades são indispensáveis na construção do conteúdo e, principalmente, na sua organização e estruturação em níveis. A construção do conteúdo e seu dimensionamento devem ser pensados de modo que possa contribuir para o processo de ensino aprendizagem, na transposição didática de conteúdos.

Para este trabalho optou-se por utilizar o assunto Introdução ao Processamento de Dados (IPD), desenvolvido por Melo (MELO et al, 2005), por ser

mais abrangente e se adequar mais ao perfil dos estudantes que participaram da coleta de dados. O curso em questão possui um conteúdo disposto de acordo com a estrutura do tutor, com quinze unidades didáticas conceituais, organizadas em sequência lógica e dispostas em cinco níveis, facilitado, médio, avançado, exemplos e respostas a perguntas frequentes (FAQs).

5.1.2 Recursos computacionais utilizados neste trabalho

Os sistemas foram hospedados em uma solução de computação nas nuvens (*cloud computing*), baseada no sistema Xen¹ que permite virtualizar os servidores com alto desempenho e alocar recursos individualmente. Para este trabalho, utilizou-se a alocação dos recursos virtuais com as características apresentadas no quadro da Tabela 5.1

Tabela 5.1: Quadro de recursos computacionais utilizados neste trabalho

Processador	2 vCPUs
Memória RAM	1 GB
Disco Rígido	50 GB
Transferência Mensal	175 GB
Sistema Operacional	Linux Ubuntu Server 11.10
Ferramentas de Programação	Ruby com IDE IntelliJIdea, banco de dados Postgres e servidor web Mongrel

¹ Xen é um software livre que tem a função de emular sistemas operacionais, fazendo a virtualização para as arquiteturas x86, x86-64, IA-32, IA-64 e PowerPC.

5.1.3 Definição das regras especialistas

No paradigma de Inteligência Artificial Simbólica, muitas técnicas têm sua estruturação no conhecimento do especialista. O especialista apresenta-se como um indivíduo ou um grupo de indivíduos com domínio de um determinado assunto e com capacidade de fazer relações, estabelecer fatos, regras e procedimentos que subsidiem na modelagem do conhecimento. Para o tutor com orientação especialista, utilizado neste trabalho, foram utilizadas as regras extraídas do trabalho realizado por Melo et al (2005). Portanto, as regras especialistas seguem o padrão apresentado anteriormente na Figura 4.7 do capítulo 4 deste trabalho.

5.1.4 Coleta de dados nos tutores especialista e híbrido

Com o objetivo de obter maior uniformidade nos dados, tanto nos componentes do conjunto de treinamento quanto do tutor híbrido, a coleta foi limitada a uma população específica. Foram escolhidos alunos dos primeiros anos dos cursos Técnicos Integrados em Química, em Informática e em Mecânica, todos do Instituto Federal de Goiás, Campus Luziânia. A seleção do público alvo baseou-se entre outros fatores, no conteúdo do tutor e na necessidade de que os participantes possuíssem conhecimentos básicos de informática.

Após a seleção do público alvo, procedeu-se a escolha de um ambiente controlado para a coleta de dados. Para este caso, optou-se por utilizar o Laboratório de Informática do Instituto Federal de Goiás (IFG), Sala S401B, Campus Luziânia, Goiás, Brasil.

Para possibilitar uma familiarização com o ambiente foi desenvolvido um curso intitulado Teste de Navegação no Sistema, com uma única unidade didática, porém, composto por cinco níveis distintos: facilitado, médio, avançado, FAQ e exemplos, assim como as avaliações inicial e final. Com isso, antes de iniciar a coleta de dados, os estudantes foram convidados a fazer o treinamento no próprio ambiente, em um curso cujo conteúdo versava sobre a utilização do próprio sistema. Assim, nesse treinamento foram apresentadas todas as etapas de utilização do tutor, bem como os cuidados e as exigências no processo de coleta de dados. Adicionalmente, fez-se a apresentação das principais características do tutor com a finalidade de minimizar os problemas inerentes à navegação. Ainda, foram transmitidas informações sobre o software, tais como a estrutura, o ambiente de desenvolvimento e o tempo médio para a conclusão do curso a ser estudado.

As coletas nos tutores com orientação especialista e híbrido foram realizadas com dois grupos diferentes, um para cada tutor. Esse procedimento permite a comparação de grupos com características comuns, mas submetidos a situações distintas.

5.1.5 Conjunto de treinamento

Para compor o conjunto de treinamento foram realizadas três coletas de dados utilizando o tutor com orientação especialista, totalizando trinta e dois estudantes. Durante a coleta do último grupo, o *link* de internet apresentou problemas e a coleta foi interrompida, sendo reiniciada no dia seguinte. Apesar do sistema poder recomeçar o curso do local em que foi interrompido, decidiu-se desconsiderar os

dados desse grupo para a composição do conjunto de treinamento, por receio de que a descontinuidade pudesse gerar padrões com ruídos. Sendo assim, somente os dados de vinte e dois estudantes foram selecionados para a composição do conjunto de exemplos.

Os testes inicial e final, compostos de quinze questões cada, produziram 660 (seiscentos e sessenta) registros. Em cada um existem informações da navegação, incluindo o nível de acerto, latência (tempo em milissegundos) e grau de satisfação.

O curso utilizado na coleta de dados possui quinze unidades didáticas, divididas em cinco níveis cada (facilitado, médio, avançado, FAQs e exemplos), onde cada um desses níveis possui um exercício associado. As visitas a essas unidades e seus respectivos exercícios produziram 624 (seiscentos e vinte e quatro) registros de navegação em conteúdos e o mesmo número de navegação nos exercícios, totalizando 1248 (um mil duzentos e quarenta e oito) exemplos. O gráfico da Figura 5.1 mostra a distribuição de visitas por unidades didáticas e níveis durante a coleta de dados. As unidades didáticas do curso, conforme a ilustrado no gráfico da Figura 5.1, são constituídas de conteúdos sobre: codificação da Informação, CPU, Estrutura do sistema computacional, Hardware, Informática, Introdução, Memória, Multimídia, Periféricos, Processamento de Dados, Programação de computadores, Programas Aplicativos, Sistema Operacional, Software e Tipos de computadores.

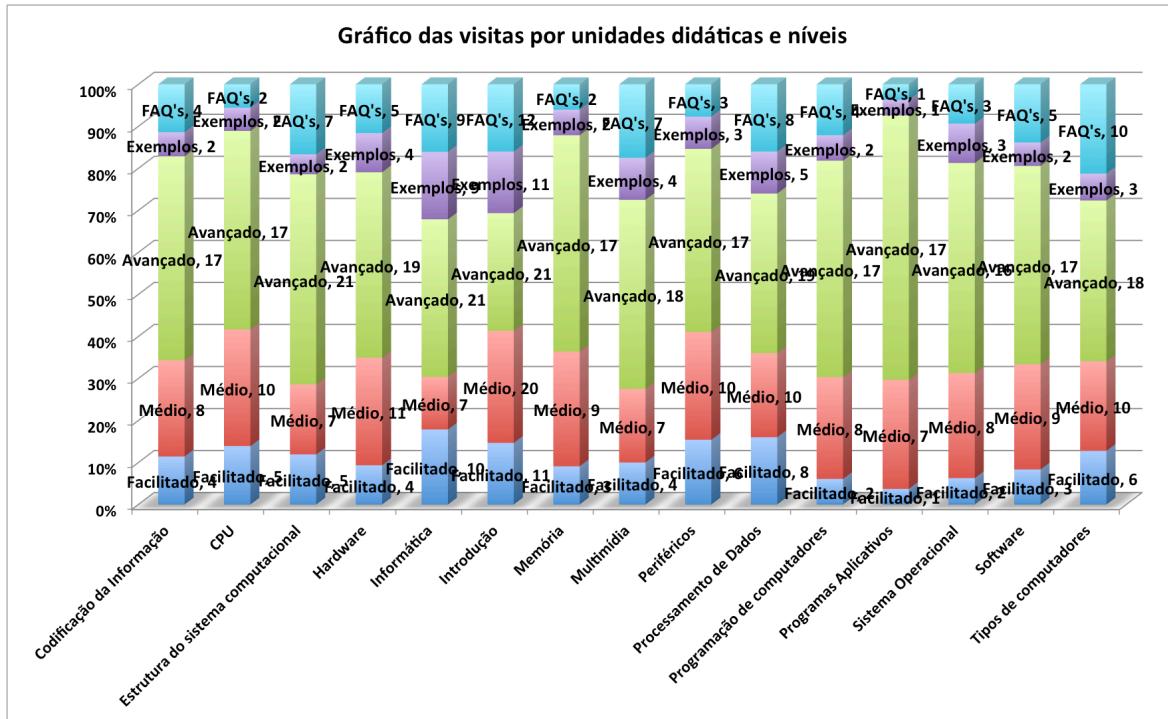


Figura 5.1: Visitas por unidades didáticas e níveis.

Em cada registro de navegação foram armazenados os seguintes dados: latências (em milissegundos), grau de satisfação, sequência ou ordem de visita e nível de acerto (quando exercício), além dos dados que permitem identificar o registro.

Neste trabalho, pressupõe-se que a decisão de qual o próximo nível ou unidade didática a estudar é afetada pela situação atual, uma vez que ela terá influência no passo seguinte do estudante. Isso ocorre porque o tutor é capaz de perceber o comportamento da navegação do estudante, instantaneamente, e então orientá-lo para o conteúdo mais adequado ao desempenho apresentado naquele instante.

Assim, os dados que compõem o conjunto de treinamentos da rede foram limitados àqueles em que a decisão atual do estudante (tempo t) tenha influ-

enciado positivamente no resultado obtido no próximo item estudado (tempo $t + 1$). Ou seja, são utilizados para o treinamento apenas as decisões que tenham proporcionado um nível de acerto do próximo nível ou da unidade didática estudada.

Com a seleção dos dados, o número de registros qualificados para compor o conjunto de exemplos utilizado no treinamento das redes neurais reduziu para 924 (novecentos e vinte e quatro). Desses, 462 (quatrocentos e sessenta e dois) são informações de navegação em unidades didáticas e o mesmo número de registros de navegação em exercícios. Além dos registros em que a decisão atual não tenha resultado em acerto no passo seguinte, excluiu-se também os últimos registros de cada estudante, pois eles sempre conduzem ao teste final do tutor.

5.1.6 Conjunto de validação

Para a validação dos resultados, procedeu-se a uma nova coleta de dados utilizando-se o tutor híbrido, respeitando os mesmos critérios adotados na coleta de dados com o tutor especialista. Assim, manteve-se o mesmo perfil de público, com duas coletas de dados, totalizando vinte e dois estudantes.

Descritos os dados para a definição do conjunto de treinamento das redes e da validação é apresentado a seguir o processo de treinamento automatizado.

5.2 Avaliação do treinamento automatizado

Uma das contribuições desta tese é o treinamento automatizado, desenvolvido com o objetivo de possibilitar a utilização do tutor em diferentes temas, sem que seja necessário fazer adaptações em sua estrutura.

A primeira experiência com a implementação do algoritmo foi realizada utilizando-se o conceito de orientação a objetos. Todavia, com esse paradigma de programação, o treinamento de todas as redes do curso utilizado no experimento foi estimado em aproximadamente cinquenta e cinco dias. Com o objetivo de tornar o processo de treinamento mais ágil, toda a estrutura foi reconstruída com o uso de vetores, utilizando o paradigma estruturado de programação. Esse formato, permitiu reduzir o tempo de treinamento para aproximadamente dezoito horas.

Todo o processo de treinamento foi realizado em ambiente web no mesmo servidor mencionado no quadro da Tabela 5.1. É provável que a implementação do algoritmo de treinamento em uma linguagem compilada, de maior desempenho e que utiliza menos recursos computacionais, por exemplo a linguagem C, poderá reduzir seu tempo de execução.

Observando o algoritmo apresentado na Figura 4.8 e aplicando-o ao conjunto de treinamento obteve-se o que segue.

Para definir as dimensões das redes a serem treinadas, inicialmente estima-se a quantidade de padrões que cada neurônio deverá representar, considerando números que possibilitem a generalização das redes. Para esse experimento essa quantidade foi limitada entre sete e vinte padrões por neurônio, o que resultou em redes com até 66 neurônios (462 padrões divididos por 7 neurônios por padrão) e redes com 23 neurônios (462 padrões divididos por 20 neurônios por padrão).

O passo seguinte foi definir as dimensões máximas e mínimas das redes bidimensionais, extraindo-se a raiz quadrada do total de neurônios estimados

para a rede. Para esse caso as raízes quadradas foram 8 e 4, por aproximação. Logo, a dimensão máxima foi definida em 8 e a mínima em 4. O quadro da Tabela 5.2 mostra o resultado do treinamento automatizado das redes neurais.

Tabela 5.2: Quadro das redes do treinamento automatizado

Dimensão da rede	Total de neurônios	Qualidade Topológica	Qualidade de Precisão
4	16	126	42,83
5	25	278	31,45
6	36	379	23,71
7	49	247	18,87
8	64	435	15,00

O sistema treinou as cinco redes e selecionou para o curso do experimento a rede com dezesseis neurônios, que apresentou melhor ordenação topológica. Despois de concluído o treinamento, foi realizada a coleta de dados no tutor híbrido.

Apresentado o treinamento automatizado a seguir são analisados os resultados obtidos nos tutores.

5.3 Análise dos resultados dos tutores

A coleta no tutor com orientação especialista resultou em um conjunto de dados que além de ser a base para o treinamento da rede neural utilizada no sistema híbrido também é objeto de estudo para a avaliação e a validação dos experimentos. Por sua vez, a coleta de dados no tutor híbrido é utilizada para a comparação entre o modelo híbrido proposto e o modelo especialista. Os resultados obtidos nos tutores são apresentados a seguir.

5.3.1.1 Resultados dos testes iniciais e finais

A finalidade do teste inicial é avaliar o conhecimento prévio que o estudante tem do conteúdo do tutor. O resultado dos testes iniciais é diretamente influenciado pelo perfil do público alvo utilizado na coleta.

Para efeito de comparação, também é importante conhecer o desempenho dos estudantes ao final do curso quando submetido a outro conjunto de exercícios, neste trabalho denominado de teste final. A Figura 5.2 mostra os resultados obtidos nos testes inicial e final, nos tutores com decisão especialista e híbrido.

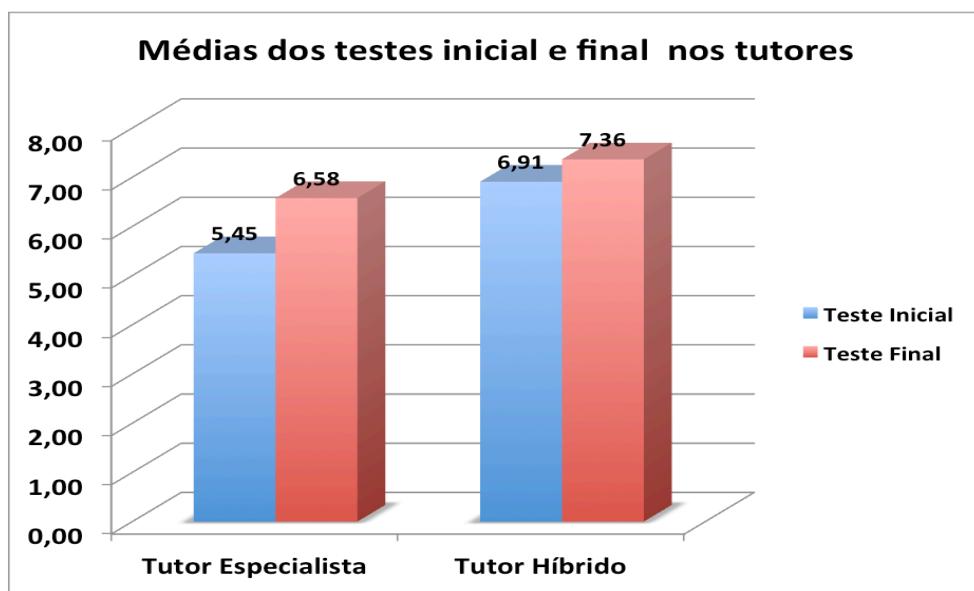


Figura 5.2: Avaliação dos testes iniciais e finais nos tutores.

Pode-se observar no gráfico da Figura 5.2 que em ambos os tutores os estudantes foram capazes de ampliar a média de questões respondidas corretamente ao final do curso.

Em uma análise mais criteriosa, pode-se verificar que a proposta de deslocamento de nível entre as unidades didáticas possibilita que os estudantes se-

jam submetidos sempre ao nível de conteúdo mais adequado ao seu perfil. Isso pode ser considerado como uma contribuição deste modelo. Por outro lado, ao realizar o deslocamento muitas vezes os estudantes podem não ser encaminhados aos níveis médios. Isso ocorre porque o desempenho de determinado estudante pode conduzi-lo com maior frequência a um nível facilitado, ou mesmo avançado. Todavia, no projeto original dos tutores, os testes iniciais e finais foram construídos com base nos níveis médios das unidades didáticas. Acredita-se que essa estratégia possa ter interferido para que o resultado dos testes finais nos tutores não fossem tão significativos. Mesmo assim, os resultados apresentaram uma melhora nas médias que aumentou no tutor especialista de 5,45 nos testes iniciais para 6,58 nos testes finais e no modelo híbrido de 6,91 nos testes iniciais para 7,36 nos testes finais. Em estudos futuros o modelo deverá ser adaptado para também possibilitar uma avaliação final adaptativa conforme o perfil de cada estudante.

A variação dos testes iniciais e finais também foi avaliada, como mostrado no gráfico da Figura 5.3.

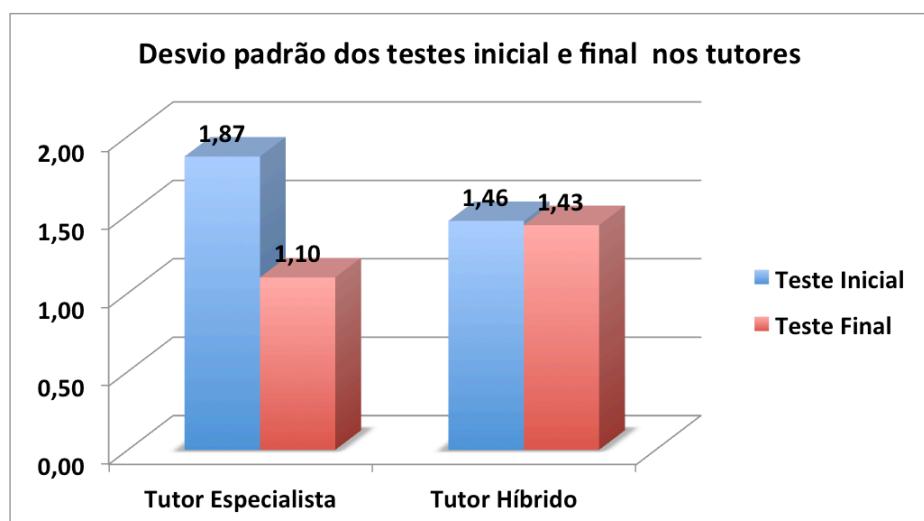


Figura 5.3: Desvio padrão dos testes iniciais e finais nos tutores.

Como pode ser observado no gráfico da Figura 5.3, os desvios padrões no tutor especialista foram de 1,87 no teste inicial e 1,10 no teste final. No tutor híbrido, os desvios foram de 1,46 no teste inicial e de 1,43 no teste final. Os resultados apresentaram uma diminuição na variação das médias obtidas pelos estudantes nos testes finais, quando comparados aos testes iniciais dos tutores.

Descritos os resultados dos testes inicial e final, a seguir são apresentados os resultados obtidos pelos estudantes nas resoluções dos exercícios das unidades didáticas.

5.3.1.2 Análise do resultado dos exercícios das unidades didáticas

Como mostrado no Capítulo 4 deste trabalho, a navegação no tutor com orientação especialista é orientada pela decisão do professor que desenvolveu o curso. No entanto, nesse modelo é apresentada uma tela de navegação ao estudante, que contém além da unidade didática indicada pelo especialista os outros níveis disponíveis e ainda não estudados do mesmo conteúdo. Sendo assim, o estudante tem a liberdade de escolher o nível que seja mais adequado à sua forma de estudo, ou mesmo uma nova unidade didática.

Quando um aluno decide estudar uma nova unidade didática, seja por interesse próprio ou por ter estudado todos os níveis daquele conteúdo, ele é encaminhado para o nível da próxima unidade didática que seja mais adequado ao seu desempenho global. Assim, dependendo do seu desempenho o aluno pode ser encaminhado aos níveis facilitado, médio ou avançado de uma nova unidade didática.

Este trabalho convencionou que estudantes com aproveitamento superior a 75% na avaliação inicial começariam o curso em um nível avançado. Por outro lado, aqueles que tivessem um aproveitamento inferior a 25% iniciariam o curso em um nível facilitado. Os demais que tivessem um aproveitamento no intervalo entre 25% e 75% iniciariam o curso no nível médio. O mesmo percentual de aproveitamento é utilizado em relação às mudanças de conteúdo ou unidade didática. No entanto, o desempenho global para a navegação entre as unidades didáticas não é medido pelo resultado da avaliação inicial, mas sim pelo desempenho obtido nas respostas aos exercícios das unidades didáticas já estudadas.

Com a estrutura desenvolvida para os tutores com orientação especialista e híbrida, o estudante tem a opção de navegar por quaisquer dos níveis das unidades didáticas ou, no pior dos casos, em pelo menos um nível de cada assunto.

A Figura 5.4 mostra o gráfico dos resultados obtidos nos exercícios das unidades didáticas do tutor com orientação especialista.

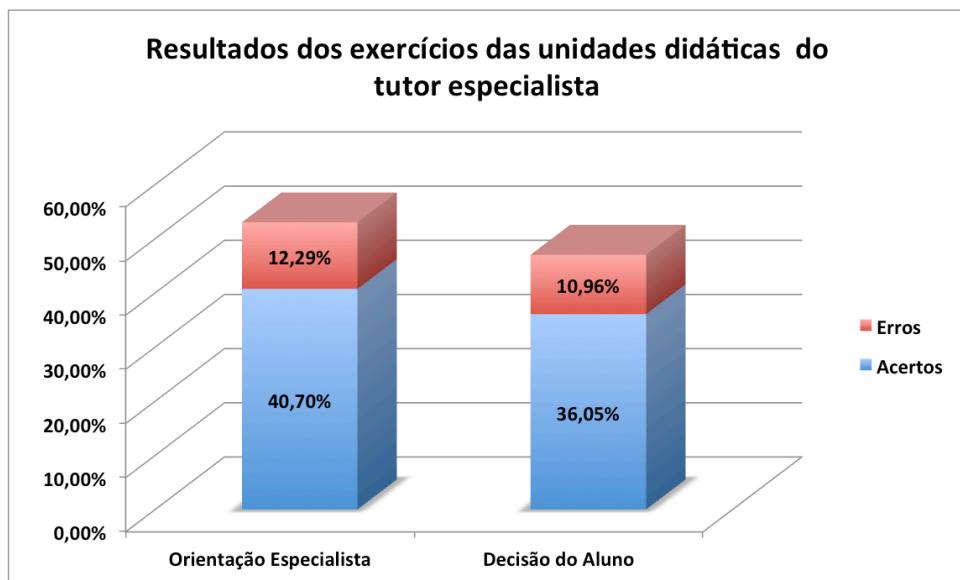


Figura 5.4: Repostas a exercícios das unidades didáticas do tutor especialista.

Pode-se observar no gráfico Figura 5.4 que do total de exercícios das unidades didáticas respondidos pelos estudantes 52,99% foram de níveis ou unidades didáticas sugeridas pelo professor especialista, com 12,29% de erros e 40,70% de acertos. Por outro lado, 47,01% dos estudantes tomaram suas próprias decisões, com 10,96% de erros e 36,05% de acertos.

O percentual de acerto dos estudantes que tiveram decisão própria foi de 76,68% dos exercícios e aqueles que seguiram a orientação especialista acertaram 76,80%. Com base nesses números, pode-se utilizar a estratégia de estudo do aprendiz para a construção da estratégia de ensino do sistema tutor híbrido.

Com o objetivo de comparar os resultados entre os tutores, o gráfico da Figura 5.5 mostra o comportamento da navegação no tutor híbrido.

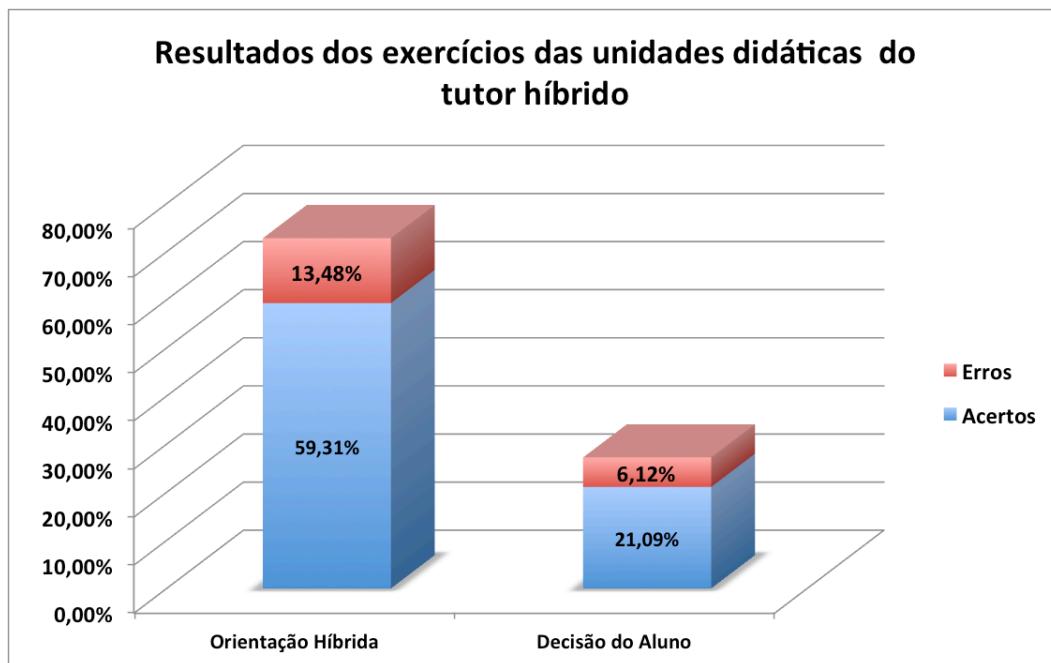


Figura 5.5: Respostas a exercícios das unidades didáticas do tutor híbrido.

Pode-se observar no gráfico da Figura 5.5 que do total de exercícios

das unidades didáticas respondidos pelos estudantes 72,79% foram de níveis ou unidades didáticas sugeridas pelo tutor híbrido, com 13,48% de erros e 59,31% de acertos. Por outro lado, 27,21% dos estudantes tomaram suas próprias decisões, com 6,12% de erros e 21,09% de acertos.

O percentual de acerto dos estudantes que tiveram decisão própria foi de 77,51% dos exercícios e aqueles que seguiram a orientação do tutor híbrido acertaram 81,48%. Portanto, na orientação híbrida ocorreu um aumento de 4,67% no nível de acertos em relação ao modelo especialista.

É importante observar que a estrutura desenvolvida para o tutor híbrido possibilita um novo treinamento, a qualquer instante, incluindo no conjunto de exemplos os dados gerados pelo próprio modelo híbrido. Com isso, é possível refinar ainda mais a estratégia de ensino do modelo.

A subseção 5.3.1.3 deste capítulo mostra que os dados também permitem avaliar o nível de aceitação das indicações dos tutores.

5.3.1.3 Avaliação da aceitação das indicações dos tutores

Ao analisar os resultados obtidos nos exercícios das unidades didáticas, pode-se verificar que aumentou a aceitação dos estudantes às indicações do tutor híbrido. O gráfico da Figura 5.6 mostra esses resultados.

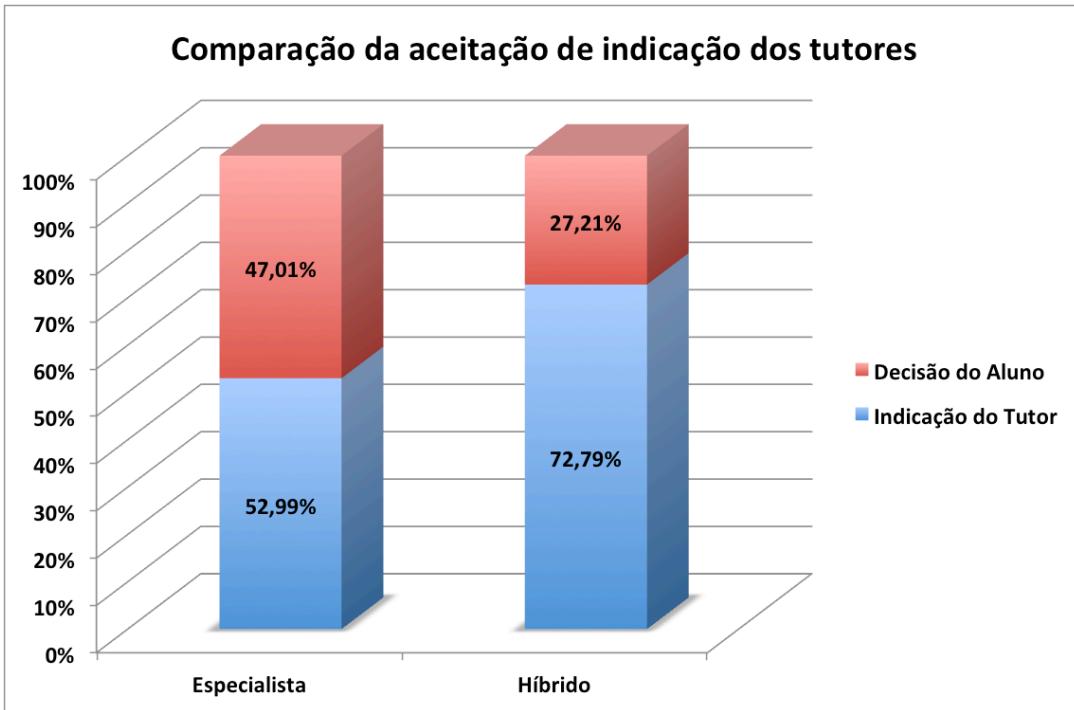


Figura 5.6: Comparação da aceitação de indicação dos tutores.

Pode-se observar no gráfico da Figura 5.6 que 72,79% dos alunos que estudaram no tutor com orientação híbrida seguiram a orientação do sistema, enquanto somente 52,99% seguiram a orientação do tutor especialista. Isto representa um aumento de 19,80% no índice de aceitação da orientação do tutor híbrido. Pode-se considerar que esse resultado é uma melhora na capacidade de adaptação do tutor híbrido ao perfil do estudante.

Durante todo o processo de navegação o estudante indica o nível de satisfação, tanto em relação ao conteúdo quanto à estrutura dos exercícios. A seguir, é apresentada uma avaliação do nível de satisfação.

5.3.1.4 Avaliação dos níveis de satisfação nos tutores

As confirmações do estudante após a conclusão da leitura de um con-

teúdo ou resolução de um exercício são realizadas em ícones que devem indicar o nível de satisfação, seja ele em relação ao conteúdo ou ao exercício. As opções disponíveis são: satisfeito, indiferente e insatisfeito. O gráfico da Figura 5.7 mostra os resultados obtidos dessas confirmações nos tutores.

Pode-se observar no gráfico da Figura 5.7 que os níveis de satisfação no tutor especialista se concentraram em satisfeito com mais de 99% de índice de aprovação. Por outro lado, no tutor híbrido ocorreu uma diminuição nesse índice, que foi de 92,48%. Em relação ao tutor especialista esse nível representou uma diminuição de 6,42%.

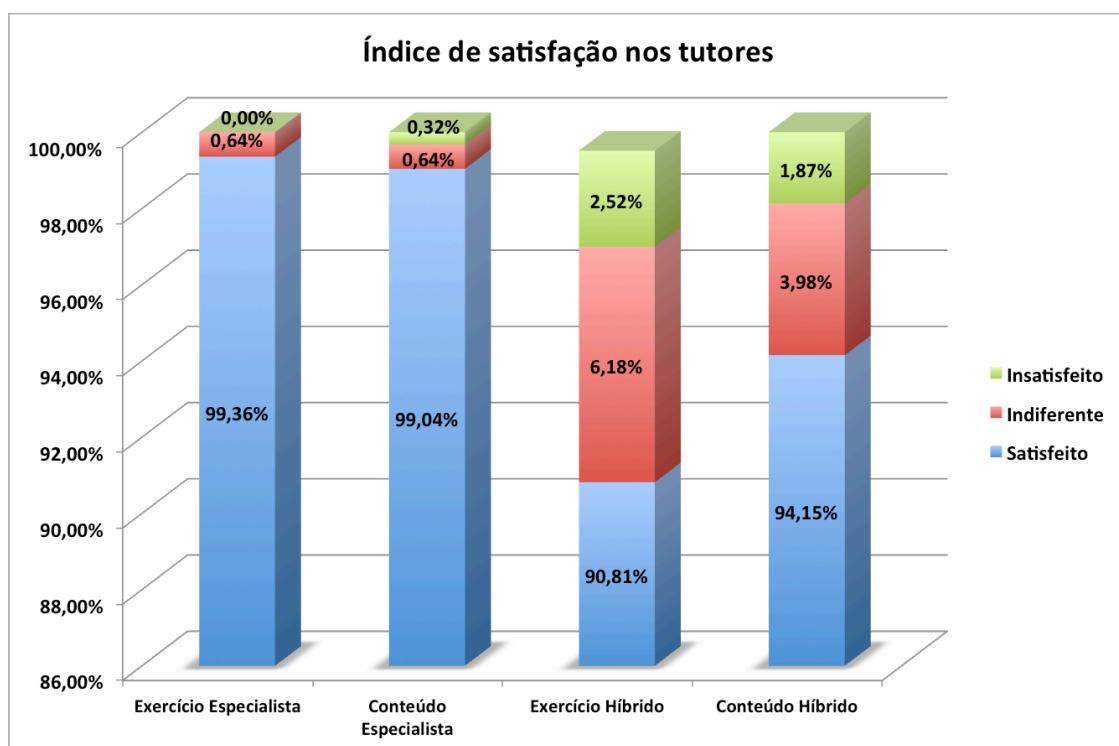


Figura 5.7: Nível de satisfação dos tutores especialista e híbrido.

É importante ressaltar que os sistemas especialista e híbrido utilizaram o mesmo ambiente para a coleta de dados e mantiveram a mesma estrutura, incluindo layout, textos e exercícios. Esse resultado deve ser objeto de estudo futuro, mas

acredita-se que a redução da satisfação pode ter sido devido à característica não linear das redes neurais, que proporciona ao sistema híbrido um maior número de orientações de níveis e conteúdos a estudar.

Este capítulo apresentou o desenvolvimento do experimento e a análise dos resultados obtidos nas coletas de dados realizadas utilizando-se os sistemas tutores com orientação especialista e híbrida.

Os resultados mostraram que os estudantes diversificam o seu perfil de navegação, possibilitando seu aproveitamento para a construção de novas estratégias de ensino. Foi observado também um aumento na retenção de conteúdo a curto prazo ou aprendizado (ALBUQUERQUE, 2001).

Os dados ainda mostraram que quando comparado ao tutor especialista ocorreu um aumento na aceitação das indicações do tutor híbrido. Ainda foi possível verificar que no modelo híbrido ocorreu uma diminuição no índice de satisfação.

Também foi apresentado o processo de treinamento automatizado, que mostrou ser capaz de adequar o modelo neural ao conjunto de dados gerados por cada curso.

Concluídas as análises dos resultados e experimentos, o próximo capítulo apresenta as conclusões, contribuições e publicações obtidas nesta tese e os trabalhos futuros que poderão ser desenvolvidos a partir desta.

CAPÍTULO 6: CONCLUSÕES E CONTRIBUIÇÕES DESTE TRABALHO E TRABALHOS FUTUROS

Ninguém é tão sábio que nada tenha para aprender, nem tão tolo que nada tenha pra ensinar. (Blaise Pascal)

Esta tese formalizou a proposta de um modelo de Sistema Tutor Inteligente Híbrido que utiliza orientação especialista subsidiada pelas redes neurais artificiais Mapas Auto-Organizáveis ou redes SOM (mapas de Kohonen).

Este capítulo apresenta as conclusões e as contribuições desta tese, os trabalhos futuros e as publicações originadas desta tese.

6.1 Conclusões

Apesar de existirem estudos desenvolvidos na área de STI, a maior parte deles tem sua estrutura baseada nos paradigmas da Inteligência Artificial Simbólica (GIRAFFA E VICCARI, 1997).

Tem-se conhecimento da aplicação de redes neurais artificiais na construção de sistemas tutores a partir do início dos anos 2000. Naquela época Alencar (2000) apresentou um estudo sobre a capacidade de generalização de pa-

drões e a consequente possibilidade de aplicação do paradigma conexionista em modelos de tutores. Outros autores apresentaram modelos de tutores utilizando redes neurais artificiais. Carvalho (2002) e Dorça et al (2009) usaram as redes SOM em seus projetos, Melo et al (2005) e Almeida (2006) utilizaram as redes MLP. Martins et al (2006) utilizaram a estrutura proposta por Carvalho (2000) com modificações em algumas variáveis do tutor, tal como a substituição dos exercícios das unidades didáticas pela aplicação de mapas mentais.

Um conjunto de tarefas antecedem o uso dos sistemas nos modelos conexionistas. Esses modelos trazem consigo os trabalhos inerentes à utilização de redes neurais como, por exemplo, a necessidade de coletas de dados, treinamento e avaliação das redes neurais, avaliação do desempenho das redes por um especialista, entre outras.

Neste trabalho desenvolveu-se inicialmente um sistema tutor especialista onde a base da estratégia de ensino são as regras de navegação estabelecidas pelo professor criador do curso. Posteriormente, desenvolveu-se um modelo de Sistema Tutor Inteligente Híbrido em que a base da orientação é a decisão do professor especialista subsidiada pelas decisões de uma rede neural SOM. Adicionalmente, foi desenvolvido um algoritmo de treinamento automatizado, responsável pelo treinamento e pela escolha da rede que mais se adapta ao conjunto de exemplos.

Assim, um curso que seja construído segundo os pressupostos do modelo híbrido poderá ser utilizado imediatamente após a sua criação. Inicialmente, o tutor tem suas reações com base no conhecimento especialista e à medida que o sis-

tema é usado esse conhecimento é gradualmente refinado pelas decisões do modelo neural. A base das decisões da rede neural são os padrões de navegação dos estudantes que utilizaram o modelo especialista.

Para obter os resultados foram realizadas coletas de dados com estudantes dos primeiros anos dos cursos Técnicos Integrados do Instituto Federal de Goiás. Os dados coletados foram utilizados para o treinamento das redes neurais, avaliação de desempenho dos estudantes e para a comparação entre os tutores especialista e híbrido.

Os dados coletados nos sistemas especialista e híbrido mostraram que o sistema proposto neste trabalho foi capaz de aprender com a forma de estudo dos alunos que utilizaram o modelo especialista.

O aumento no nível de aceitação das orientações do tutor híbrido indica que o sistema proposto é capaz de adaptar-se ao perfil do estudante ajustando a estratégia de ensino especialista com base na estratégia de estudo dos alunos, tornando-o mais adaptativo ao aprendiz

6.2 Contribuições

Este trabalho apresentou o modelo de um sistema tutor híbrido em que as decisões iniciais de orientação de estudos propostas por um professor podem ser melhoradas considerando a forma com que os estudantes navegam pelo sistema. Esse modelo possibilita o ajuste das decisões aos contextos locais e não se limita a um conjunto de regras de abrangência global e pré-definidas.

Portanto, este trabalho apresenta um novo modelo de desenvolvimento de Sistemas Tutores Inteligentes que inclui o professor no processo decisório e o auxilia na construção da estratégia de ensino. Esse modelo é capaz de ajustar possíveis distorções com base nas decisões da rede neural. Isto não significa que o modelo em questão deverá substituir quaisquer outras abordagens utilizadas na criação de STIs, mas, sim, que ele pode ser aplicado como método alternativo e até mesmo em conjunto com outras tecnologias.

Quando comparado com as propostas conexionistas apresentadas por Carvalho (2003) e Melo et al (2012) o modelo proposto neste trabalho contribui ainda com o deslocamento do eixo de decisão entre os níveis. Assim, esse modelo tornou-se ainda mais adaptativo pois permite uma navegação personalizada também entre as unidades didáticas.

O treinamento automatizado apresenta-se como uma contribuição, pois, considerando o conjunto de treinamento ele permite estabelecer um conjunto possível de redes neurais e define a rede neural que apresenta melhor qualidade de resposta. A rede neural que mais se adapta aos padrões é utilizada automaticamente no curso criado no tutor.

O STI desenvolvido nesta tese ainda tem a vantagem de poder ser utilizado para diversos cursos simultaneamente. A sua estrutura possibilita diminuir o trabalho dos envolvidos no processo de adaptação do modelo neural quando um novo curso é inserido no sistema.

6.3 Trabalhos Futuros

O desenvolvimento desta tese e a variedade de informações obtidas em todo o processo de construção dos sistemas, a coleta de dados e a análise dos resultados tornaram-se fonte de reflexão e inspiração para o desenvolvimento de novos projetos. Adicionalmente, existe, também, a possibilidade de melhorar o sistema proposto estendendo seu uso tanto no ensino presencial, quanto a distância.

Portanto, a partir deste trabalho, pode-se estabelecer como trabalhos futuros:

- reestruturar o sistema de avaliação final utilizando métodos que possibilitem maior adaptação ao perfil do estudante e a indicação dos níveis de aprendizado obtido;
- ampliar a estrutura do tutor permitindo uma quantidade diversificada de níveis com a finalidade de permitir uma maior flexibilidade na transposição didática de conteúdos;
- desenvolver um sistema de comunicação multidirecional implementado de forma automatizada e utilizando técnicas de inteligência artificial com a finalidade de auxiliar os alunos durante seus estudos; e
- aplicar o modelo com treinamentos utilizando padrões gerados pelo tutor híbrido com o objetivo de avaliar a sua capacidade de adaptação.

6.4 Publicações do autor relacionadas a este trabalho

- MELO, F. R.; FLÔRES, E. L.; CARVALHO, S. D.; MARTINS, W.; CAR-

RIJO, G. A.; VEIGA, A. C. P. **Multilevel didactic content for reactive personalization on intelligent tutor systems.** In: SBA - Revista da Sociedade Brasileira de Automática. São Paulo: UNICAMP, 2012.

- CARVALHO, S. D.; FLÔRES, E. L. ; MELO, F. R.; LOJA, L. F. B.; CARNEIRO, M. B. P.. **Sistema Tutor Inteligente Híbrido com Transposição Didática de Conteúdo Orientada por conhecimento Especialista e Mapas Auto Organizáveis.** In: X CEEL - Conferência de Estudos em Engenharia Elétrica, 2012, Uberlândia. Anais da Conferência de Estudos em Engenharia Elétrica, 2012. p. 30-42.
- MELO, F. R.; FLÔRES, E. L.; CARVALHO, S. D.; CARNEIRO, M. B. P.; LOJA, L. F. B. . **Redes Neurais Artificiais na Classificação de Padrões Proximais de Aprendizagem.** In: X CEEL - Conferência de Estudos em Engenharia Elétrica, 2012, Uberlândia. Anais da Conferência de Estudos em Engenharia Elétrica, 2012. p. 10-22.
- MELO, F. R.; FLÔRES, E. L.; CARVALHO, S. D. **Multilevel content's structure for personalization in conexionist intelligent tutor systems.** In: 8th International Conference on Information Systems and Technology Management, 2011, São Paulo. Abstract and Proceedings of 8th CONTECSI. São Paulo: TECSI EAC FEA USP, 2011.
- MELO, F. R.; FLÔRES, E. L.; CARVALHO, S. D.; CARRIJO, G. A.; VEIGA, ANTÔNIO CLAUDIO PASCHOARELLI . **Estruturação Reativa de Conteúdos por Diferenças Finitas em Sistemas Tutores Inteligentes Co-**

nexionistas. In: IX Conferência de Estudos em Engenharia Elétrica, 2011, Uberlândia. IX CEEL. Uberlândia: CEEL UFU, 2011.

- MELO, F. R.; CARVALHO, S. D.; FLÔRES, E. L.; MARTINS, W.; PEREIRA, R. C.; CARRIJO, G. A.; VEIGA, PASCHOARELLI, A.C. **Conteúdos didáticos personalizados por padrões proximais de aprendizagem em sistemas tutoriais inteligentes.** In: VIII CEEL, 2010, Uberlândia. VIII Conferência de Estudos em Engenharia Elétrica. Uberlândia: PET Elétrica - UFU, 2010.
- CARVALHO, S. D.; MELO, F.R; MARTINS, W.; NALINI, L. E. G.; FLÔRES, E. L.. **Sistemas Tutores Inteligentes Baseados em Mapas Auto-Organizáveis.** In: Conferência de estudos em engenharia Elétrica, 2008, Uberlândia - MG. Conferência de estudos em engenharia Elétrica 2008, 2008.

6.5 Publicações submetidas e em revisão

- CARVALHO, S. D.; FLÔRES, E. L. ; MELO, F. R.; LOJA, L. F. B.; CARNEIRO, M. B. P.. **Sistema Tutor Inteligente Híbrido Utilizando Conhecimento Especialista e Mapas de Kohonen.** In: SBA - Revista da Sociedade Brasileira de Automática. São Paulo: UNICAMP, 2012. (submetido em Setembro de 2012, parecer com aceite parcial, aguardando versão com atualizações solicitadas pelos revisores).
- CARVALHO, S. D.; FLÔRES, E. L. ; MELO, F. R.; LOJA, L. F. B.; CAR-

NEIRO, M. B. P.. **Hybrid Intelligent Tutoring System With Didactic Transposition Of The Subjects Guided By Expert Knowlegdment And Self Organizing Maps Neural Network.** In: IEEE Latin America Transactions, 2012. (submetido em Setembro de 2012, aguardando parecer dos revisores).

Este capítulo apresentou as conclusões e as contribuições desta tese, os trabalhos futuros e as publicações dela originadas.

REFERÊNCIAS

- ALBUQUERQUE, E. P. V. **Memória implícita e processamento.** Centro de Estudos em Educação e Psicologia, Instituto de Educação e Psicologia, Universidade do Minho, 2001.
- ALENCAR, W. S. **Sistemas tutores inteligentes baseados em redes neurais.** Dissertação (Programa de Pós-Graduação em Engenharia Elétrica e de Computação). Universidade Federal de Goiás, Goiânia, 2000.
- ALMEIDA, M.C. **Sistema tutor monitorado por rede neural artificial.** Dissertação (Programa de Pós-Graduação em Engenharia Elétrica). Universidade Federal de Uberlândia, Uberlândia-MG, 2006.
- CARVALHO, A.C.P.L.F. et al. **Fundamentos de redes neurais artificiais.** NCE-UFRJ. Rio de Janeiro. 1998.
- CARVALHO, S.D., **Mapas auto-organizáveis aplicados a sistemas tutores inteligentes.** Dissertação (Programa de Pós-Graduação em Engenharia Elétrica e de Computação). Universidade Federal de Goiás, Goiânia, 2002.
- CHAIBEN, H. **Um ambiente computacional de aprendizagem baseado em redes semânticas.** Dissertação (Mestrado em Ciências). Centro Federal de Educação Tecnológica do Paraná. Curitiba, 1996.
- CHEVALLARD, Y. **La transposición didáctica del saber sabio al saber enseñado.** Tradução de Claudia Gilman. 3.ed. Buenos Aires: Aique. 196p., 1998.
- CORREA FILHO, M. **A arquitetura de diálogos entre agentes cognitivos distribuídos.** (Tese de Doutorado). Rio de Janeiro: COPPE/UFRJ, 1994.
- DORÇA, F.A.; LOPES, C.; FERNANDES, M. **A multiagent architecture for distance education systems.** In Advanced Learning Technologies. Proceedings. The 3rd IEEE International Conference on, pages 368–369. IEEE, 2003.
- DORÇA et al, **Adaptativity supported by neural networks in web-based educational systems.** Journal of Education, Informatics and Cybernetics, Vol 1 (2), 2009.

DORÇA et al. **Automatic student modeling in adaptive educational systems through probabilistic learning style combinations: a qualitative comparison between two innovative stochastic approaches.** Journal of the Brazilian Computer Society, Springer, July, 2012.

GIRAFFA, L.M.M. **Fundamentos de teorias de ensino-aprendizagem e sua aplicação em sistemas tutores inteligentes.** Instituto de Informática-UFRGS. Porto Alegre. Novembro, 1995.

GIRAFFA, L.M.M; VICCARI, R.M. **The use of agents techniques on intelligent tutoring systems.** Instituto de Informática-PUC/RS. Porto Alegre, 1997.

HORTON, W.K., **Designing web-based training.** Wile, New York, New York: 2000.

JONASSEN, D.H., WANG, S. **The physics tutor: integrating hypertext and expert systems.** Journal of Educational Technology Systems, Vol. 22(1), pp. 19-28, 1993.

KOHONEN, T. **Self-organizing maps.** Berlim: Springer, 2001.

KOHONEN T. **Analysis of a simple self-organizing process. biological cybernetics.** 44:135-140. Springer, 1982.

KOHONEN T. **Self-organizing maps.** Berlim. Springer-Verlag, 2nd Ed., 1997.

LUCENA, M W. F. P. **O uso das tecnologias da informática para o desenvolvimento da educação.** Rio de Janeiro: COPPE/UFRJ, 1994.

MARKOWSKA-KACZMAR, U; KWASNICKA, H; PARADOWSKI, M. **Intelligent techniques in personalization of learning in e-learning systems.** Studies in Computational Intelligence, Volume 273/2010, 1-23, Springer, 2010.

MARTINS, W. ; CARVALHO, S. D. **An intelligent tutoring system based on self-organizing maps - design, implementation and evaluation.** Lecture Notes in Computer Science, Berlin, v. 3220, p. 573-579, 2004.

MARTINS, W. et al. **The use of conceptual maps in a hybrid intelligent tutoring system.** In: XVII Simpósio Brasileiro de Informática na Educação, 2006, Brasília. Anais do XVII Simpósio Brasileiro de Informática na Educação. Brasília : Gráfica e Editora Positivo Ltda / Sociedade Brasileira de Computação, 2006. v. 1. p. 467-476

MARTINS, F; AFONSECA, U.R. **Tutoriais inteligentes baseados em aprendizado por reforço: concepção, implementação e avaliação empírica.** Anais do SBIE 2007.

MCCULLOCH, W. S. ; PITTS, W. H. **A logical calculus of the ideas immanent in neural nets.** Bulletin of Mathematical Biophysics, v.5, 1943.

MELO, F. R. et al. **Hybrid intelligent tutoring systems based on psychological profiles and learning styles design, implementation and evaluation.** In: VII Congresso Brasileiro de Redes Neurais, 2005, Natal. Anais do VII Congresso Brasileiro de Redes Neurais, 2005. v. 1. p. 235-240.

MELO, F. R. et al. **Multilevel didactic content for reactive personalization on intelligent tutor systems.** In: SBA - Revista da Sociedade Brasileira de Automática. São Paulo: UNICAMP, 2012.

PARK, O., PEREZ, R.S.; SEIDEL, F.J., **Intelligent cai: old wine in new bottles or a new vintage ?,** em KEARSLEY, G. Artificial Intelligence and Instruction - Applications and Methods, 1987, pp. 11-45.

PARK, O. **Functional characteristics of intelligent computer-assisted instruction: intelligent features.** Educational Technology, June 1988, pp. 7-14.

RAABE, A. L. A.; GIRAFFA, L. M. M . **Uma arquitetura de tutor para promover experiências de aprendizagem mediadas.** In: XVII SImpósio Brasileiro de Informática na Educação - SBIE2006, 2006, Brasília - DF. Anais do XVII SImpósio Brasileiro de Informática na Educação, 2006. v. 1. p. 589-598.

RUMELHART, D.E. ; HINTON G.E. ; WILLIAMS, R.J. **Learning representations by back-propagation errors.** Nature, n.323 p.533-536, 1986.

ROSENBLATT, F. **The perceptron: a probabilistic model for information storage and organization in the brain.** Psychological Review, v. 65, 1958.

RUSSELL, S. J. e NORVIG, P. **Inteligência artificial.** Tradução da Segunda Edição. Rio de Janeiro: Editora Campus, 2004.

SAVIANI, D. **Pedagogia histórico crítica: primeiras aproximações.** São Paulo: Editora Cortez, 1991.

SEELEY, ROD R.; TRENT D. STEPHENS; PHILIP TATE. **Essentials of anatomy & physiology.** Segunda edição. St. Louis: Mosby-Year Book, Inc, 1996.

SHOHAM, Y. **Agent-oriented programming.** Berlin: Springer Veralg, v.60,1993.

SILVA, J.C.M. **“Ranking” de dados multidimensionais usando mapas auto-organizáveis e algoritmos genéticos.** Dissertação (Programa de Pós-Graduação em Engenharia da Computação). Universidade Federal de Goiás, Goiânia, 2000.

TARDIF, M. **Saberes docentes e formação profissional.** 4^a ed. Rio de Janeiro:

Vozes, 2002.

TAYLOR, R.P. **The computer in school: tutor, tools, tutee.** New York: Teachers College Press, 1980.

VICCARI, R.M. **Um tutor inteligente para a programação em lógica-idealização, projeto e desenvolvimento.** Tese de Doutorado (Universidade de Coimbra), 1990.

VICCARI, R.M. & GIRAFFA, L.M.M. **Sistemas tutores inteligentes: abordagem tradicional vrs. abordagem de agentes.** XII Simpósio Brasileiro de Inteligência Artificial. Curitiba. Outubro, 1996.