

Uma Aplicação da Teoria Sociointeracionista de Vygotsky para construção de um Modelo de Aluno

Adja Ferreira de Andrade¹, Lúcia M. M. Giraffa¹, Rosa Maria Vicari²

¹ Faculdade de Informática - PUCRS. Av. Ipiranga, 6681 - Prédio 16 90619-900 . Fone: +55 51 33203611 ramal 32, Fax: 3320-3621. Porto Alegre-RS.

² Pós Graduação em Informática na Educação - PGIE- UFRGS. Caixa Postal: 5071 - CEP: 90041-970. Fone: +55 51 3316 3986. Porto Alegre-RS

{adja, giraffa}@inf.pucrs.br, rosa@inf.ufrgs.br

Resumo: O objetivo deste artigo é descrever uma aplicação da teoria sociointeracionista de Vygotsky para a construção do modelo de aluno. Utiliza-se como aporte da Ciência da Educação as principais idéias envolvendo mediação, zona de desenvolvimento proximal e formas de suporte referenciadas na teoria sociointeracionista de Vygotsky. Sob o ponto de vista de Ciência da Computação buscou-se utilizar a abordagem envolvendo agentes pedagógicos dentro de ambientes de ensino-aprendizagem. A possibilidade de utilizar habilidade, competência e confiança como parâmetros do modelo do aluno mostrou-se uma abordagem interessante para definir o diagnóstico cognitivo. Do ponto de vista da Ciência da Computação, este trabalho amplia o conjunto de resultados no que concerne às potencialidades da arquitetura BDI, que se mostra bastante promissora para a modelagem de agentes pedagógicos.

Palavras-Chave: Modelagem de Aluno, Agentes Pedagógicos, teoria de Vygotsky, ZDP.

Abstract: This paper presents Vygotsky sociointeractionist theory application to build a Student Model. This approach uses the concept of Zone of Proximal Development – ZPD -, mediation and support, based on Vygotsky theory. From a Computer Science point of view this work uses an approach of pedagogical agents in educational environment. A multiagent architecture has been proposed to support the functionalities and requirements associated to a computational environment. The possibility of using skills, competencies and confidence to parameterize the student model is a new approach to define cognitive diagnosis. The computational contribution of this work is to enhance the set of results regarding the BDI architecture potentialities. This approach shows very suitable to model pedagogical agents.

Key-Words: Student Modelling, Pedagogical Agents, Vygotsky theory, ZDP.

1. Introdução

Este artigo apresenta um modelo de diagnóstico cognitivo para um ambiente de aprendizagem usando a abordagem sociointeracionista de Vygotsky. Este modelo é resultado de um estudo multidisciplinar na área Ciência da Educação e Ciência da Computação (Inteligência Artificial).

Sob o ponto de vista da Ciência da Educação, este trabalho utiliza o referencial teórico oriundo das idéias de Vygotsky (1998). Em especial, no que concerne à questão de análise da ZDP (Zona de Desenvolvimento Proximal do aluno). A noção de ZDP foi proposta por Vygotsky (1998) como sendo “a distância entre o atual nível de desenvolvimento do aluno determinado pela resolução de problemas autonomamente e, o nível de desenvolvimento potencial determinado pela capacidade de resolução de problemas sob a orientação de um adulto ou em colaboração com pares mais capazes”. A questão da mediação na ZDP e suporte aos alunos são assuntos centrais neste artigo. O suporte a ser fornecido para o aluno pode ser tanto computacional (através dos agentes), quanto humano, considerando que um aluno mais capacitado, um professor, um monitor possam auxiliar determinado aluno com dificuldade em uma atividade ou problema.

Sob o ponto de vista da Ciência da Computação, este trabalho buscou oferecer resposta a questão de como desenvolver um modelo computacional para tratar a mediação em um ambiente virtual de

aprendizagem, através da utilização da tecnologia de agentes. Propôs-se uma arquitetura multiagente para atender ao conjunto de funcionalidades e requisitos associados ao ambiente computacional que contemple as funções semiótica, diagnóstico e mediação, às quais Vygotsky descreve como importantes dentro do processo de interação. O agente cognitivo denominado de agente diagnóstico, integrante desta arquitetura multiagente, é tratado de forma detalhada neste artigo. A formalização do agente diagnóstico, o processo de raciocínio e a tomada de decisões foi feita através da arquitetura BDI, representando as atitudes de crenças (Beliefs), desejos (Desires) e intenções (Intentions). A plataforma de implementação do trabalho envolveu diversas tecnologias XBDDI (eXecutable BDI) e JAVA (maiores detalhes na seção 4).

O resultado deste trabalho é essencialmente de Informática na Educação. Isto é, não se pode separar totalmente o que é resultado de Ciência da Computação, do que é resultado para a Ciência da Educação. A justificativa das escolhas tecnológicas se fundamenta nos pressupostos da teoria educacional. A explicação do como os pressupostos inerentes da teoria educacional são aplicados encontram resposta na modelagem computacional e nas implementações realizadas.

O artigo está estruturado em cinco seções. A seção 2 apresenta alguns trabalhos correlatos que serviram de base para especificação do processo de diagnóstico. Na seção 3 detalha-se o modelo computacional do aluno tendo por base os pressupostos da teoria sociointeracionista de Vygotsky. A seção 4 descreve o agente diagnóstico em execução e o comportamento do agente especificado segundo o Modelo BDI. A seção 5 apresenta as considerações finais deste trabalho. As referências bibliográficas utilizadas para elaboração deste artigo estão colocadas no final do texto.

2. O Processo de Diagnóstico Cognitivo do Aluno

O processo de construção do agente diagnóstico iniciou-se com uma proposta de arquitetura ou sistema multiagente (Andrade et al, 2000, Andrade et al, 2001). Tal arquitetura foi inspirada em uma abordagem sociointeracionista e especifica os agentes exercendo algumas das funções (mediação, diagnóstico, suporte, etc) às quais Vygotsky descreve como importantes dentro do processo de mediação. Detalhes da arquitetura podem ser vistos em Andrade (2003).

Cabe-se inicialmente uma breve descrição dos agentes que fazem parte da arquitetura. O agente diagnóstico (Andrade, 2002a) tem como objetivo de criar o modelo de aluno a partir da análise da ZDP. Este agente cognitivo possui estados mentais utilizando a arquitetura BDI. Ele também é classificado como um agente pedagógico, porque está voltado para o ambiente de ensino-aprendizagem. Dentro da função pedagógica, ele exerce a função de um tutor e coopera com os demais agentes para auxiliar a aprendizagem a distância. O agente mediador faz a interação do aluno com o ambiente. O agente semiótico (Jung, 2001) é um agente de serviço. O agente mediador (Jaques, 2003) visa modelar os aspectos emocionais do aluno, sendo implementado enquanto um agente emocional (credível). E finalmente, a arquitetura de comunicação e interação (Quintans, 2002) responsável pela comunicação entre os agentes e a mediação do aluno com todo o ambiente.

O diagnóstico cognitivo do aluno (Andrade, 2002 b), inicia com a análise da tarefa através de um pré-teste realizado pelo aluno, a fim de identificar habilidades internalizadas (módulo *Core*) e dificuldades (módulo ZDP). O conceito de *Core* é mencionado no trabalho de Lewis (2000). Segundo o autor, “o conhecimento do indivíduo tem um núcleo central que é próprio do indivíduo. Ou seja, que ele é capaz de usar aquele conhecimento na realização de tarefas autonomamente”. Utilizando por base a teoria de Vygotsky, o *Core* representa o que Vygotsky denominou de Nível de Desenvolvimento Real (NDR), isto é, o conhecimento internalizado pelo aluno. Na verdade descreve-se o *Core* olhando para aquelas tarefas, habilidades, conceitos e ações que os alunos estão aptos a realizar e não necessitam de qualquer suporte.

A aplicação do pré-teste tem como objetivo a análise do estado corrente do conhecimento do aluno sem qualquer suporte do sistema (nível de desenvolvimento real), e o pós-teste remete à análise da interação após o suporte (nível de desenvolvimento potencial).

Após a identificação das habilidades que se encontram no *Core* (sinônimo de NDR) e na ZDP, inicia-se, efetivamente, o processo de suporte ao aluno com a sugestão de táticas adequadas ao seu trabalho. O módulo de suporte contém as táticas para auxiliar o aluno na execução da tarefa. Estas táticas podem ser constantemente modificadas e avaliadas pelo professor. Durante a construção do diagnóstico, precisa-se saber, também, o grau de confiança e competência na execução de cada habilidade, a motivação e a necessidade de suporte do aluno. Tais parâmetros irão influenciar na definição de quando uma habilidade pertence ou não ao *Core* original do aluno.

A idéia de utilizar um pré-teste para iniciar o modelo do aluno surgiu do trabalho de Luckin (1999) intitulado “*Trivar: Exploring the zone of proximal development*”. O projeto *Trivar* oferece instrução sobre ecologia a alunos na faixa etária de 10 a 11 anos de idade. Os dois objetivos do projeto são: a investigação do papel que a ZDP pode exercer para o projeto do software educacional, em especial para especificar o modelo de aluno. A autora defende a análise da zona de desenvolvimento proximal, como ingrediente essencial para uma instrução efetiva. Nesse projeto, os alunos recebem um pré-teste e depois sua performance é comparada com um pós-teste. A cada aluno é pedido para tornar explícitos os ganhos da aprendizagem no domínio de ecologia adquiridos a partir da interação com os sistemas. Segundo a autora, o ponto de contato entre os conceitos espontâneos e os conceitos científicos está na forma de colaboração dentro da ZDP. No que se refere ao modelo do aluno, o sistema mantém informações do nível de ajuda oferecido e se este nível foi adequado à realização da tarefa. Em uma outra versão do sistema são analisados os conceitos que o aluno adquiriu com sucesso, o nível de dificuldade da tarefa, a quantidade de assistência que o aluno requereu para realizar a atividade com sucesso. Estes fatores combinados são usados para analisar a ZDP do aluno. O ambiente oferece uma interface gráfica, uma base de conhecimentos (organização dos conceitos), um módulo de atividades e um conjunto de ferramentas de ajuda ao aluno. A escolha do nível de ajuda ou suporte é baseada no nível de ajuda previamente oferecido. No sistema ECOLAB (Luckin and Boulay, 1999a), os autores descrevem cinco níveis possíveis de ajuda. Se um nível de ajuda falha, o sistema oferece um nível superior. O nível cinco é indicado para alunos que precisam de maior suporte.

Em um outro trabalho “*Capability, Potential and Collaborative Assistance*”, Luckin and Boulay (1999b) defendem que o modelo do aluno é um tipo de modelo *overlay* e descrevem um conjunto de crenças do modelo de aluno: *crença habilidade*, que diz respeito à capacidade autônoma do aluno e uma *crença sobre suporte* oferecido pelo sistema. A técnica de *Overlay* (Sison and Shimura, 1998) visa criar um modelo que represente o conhecimento do especialista e um modelo que represente os conhecimentos do aluno, tentando detectar as diferenças entre ambos os modelos. Vários dos elementos citados, tanto do modelo de tarefas quanto crenças do modelo do aluno, foram incorporados ao nosso ambiente. A diferença é que o aluno também atua como parte do processo de diagnóstico, ou seja, o aluno também pode avaliar o seu grau de conhecimento e de dificuldade e a quantidade de suporte desejada para realizar a tarefa. Uma limitação do projeto TRIVAR (Luckin, 1996) é que este sistema é orientado a um domínio e a uma faixa etária específica, não podendo ser aplicado em outros contextos. Nossa proposta aqui se diferencia na medida em que o nosso sistema está aberto, a interface pode incorporar domínios variados e a análise da ZDP pode ser feita a qualquer faixa etária de alunos. Uma contribuição deste artigo que pode ser incorporada ao trabalho de Luckin (1996) é a questão de o aluno poder se auto-avaliar e também analisar o próprio diagnóstico dado pelo agente.

Em um outro trabalho, denominado *Mr. Collins* (Bull and Pain, 1995) usa o modelo de aluno aberto para inspeção e negociação pelo próprio aluno. O modelo do aluno é formado pelo perfil do estudante e informações sobre as estratégias de aprendizagem preferidas pelo aluno. Estes mesmos atributos são utilizados no modelo de aluno deste artigo, no entanto a capacidade de negociação do aluno sobre seu modelo não faz parte do escopo deste trabalho, embora o modelo seja aberto à inspeção e análise pelo aluno.

O *Mr. Collins* utiliza um sistema de confidências para a criação do modelo de aluno. A confiança do aprendiz é indicada por valores *a-d* (a sendo o mais alto valor) e a confiança do sistema na habilidade do estudante é rotulada de *1-4* (1 sendo o mais alto valor). Este mesmo grau de confiança, aliado ao grau de competência e suporte, tem sido utilizado pelo agente diagnóstico para detectar quando a habilidade do aluno se encontra tanto no *Core* (conhecimento adquirido, automatizado, internalizado) quanto na ZDP.

Dessa forma, o ambiente é capaz de comparar os resultados da interação colaborativa com os testes realizados individualmente. O sistema faz uma análise dos modelos individualmente e depois uma análise do modelo conjunto formado por estudantes em pares. O sistema contém heurísticas capazes de comparar o conteúdo de dois modelos de aluno (SM-student modelling) e a melhor maneira que estes aprendizes possuem para adquirir conhecimentos. O SM1 representa os conceitos já sabidos ou consolidados do estudante 1 e SM2, os conceitos do estudante 2. Os modelos SM1&SM2 representam o conhecimento de ambos os alunos trabalhando juntos, mas que eles não podem produzir individualmente. “Isto é similar à zona de desenvolvimento proximal proposta por Vygotsky” (Bull and Smith, 1997).

Esta mesma idéia está sendo implementada no agente diagnóstico para criação do modelo de grupo. A estratégia é detectar as habilidades do grupo, a partir da análise do modelo de cada aluno separadamente. A estratégia adotada é a da *capacidade máxima*. Ou seja, uma habilidade vai estar no *core*

do grupo se tiver no *Core* de pelo menos um dos integrantes do grupo. E vai estar na ZDP do grupo se estiver na ZDP de pelo menos um dos alunos e não estiver no core de nenhum outro aluno membro do grupo.

3. O Modelo do aluno inspirado na teoria sociointeracionista

O modelo do aluno foi inicialmente empregado como parte dos sistemas STI (Sistemas Tutores Inteligentes), juntamente com o modelo do domínio e os procedimentos de ensino. A importância do modelo do aluno dentro destes ambientes incentivou várias pesquisas na década de 80 e 90, até os dias de hoje. Na década de 80, Self (1988), chama a atenção para as diversas funções ou usos dos modelos de estudantes, em especial a função *corretiva, elaborativa, estratégica diagnóstica, preditiva e avaliativa*.

A função corretiva é utilizada para identificação do erro do aluno, armazenando uma biblioteca de possíveis erros. Esta função pode ser aplicada para remediação direta ou indireta através de “dicas” ou parte do conhecimento. Pode ser usada também para trilhar a solução, permitindo ao aluno observar seu próprio raciocínio e refletir sobre seu erro. A função elaborativa tem sido usada quando o conhecimento do estudante está correto, mas incompleto. Esta função tem auxiliado escolhas de *curriculum* ou domínios a serem oferecidos ao aluno e permitido ao aluno escolher o tópico que ele deseja aprender.

A função estratégica defende que “decisões táticas devem ser vistas como parte de algum modelo global ou plano estratégico usualmente implícito no ambiente”. Uma das classes desta função é o *modus operandi* em que as características e traços dos alunos podem mudar durante a interação. Esta atualização dinâmica do modelo do aluno pode ser feita apenas pedindo ao aluno para se autodescrever. A função diagnóstico é usada para esclarecer ambigüidades dentro do modelo do aluno, avaliar suas hipóteses sobre o conhecimento aprendido. A função preditiva faz a predição da performance do aluno e suas possibilidades de aprendizagem. Ela pode ser aplicada também para investigar os efeitos das ações didáticas dos tutores. Finalmente, a função avaliativa é o meio mais utilizado para a avaliação da performance tanto do aluno, quanto do sistema. No caso do sistema, esta função é capaz de prever as estratégias adequadas a serem aplicadas pelo tutor.

Neste artigo, o modelo do aluno utiliza a função diagnóstico para descrever alguns componentes que identifiquem o conhecimento do aluno: perfil_aluno, perfil_aprendizagem, perfil_motivacional, auto_avaliação, diagnóstico_tarefa e suporte. Na tabela 1 é descrito cada componente, os atributos contidos dentro de cada componente, a descrição, como e quando este atributo é adquirido ou atualizado e quem pode modificá-lo.

Tabela 1: Modelo do Aluno

Componente	Atributo	Descrição	Como adquirido	Quando adquirido	Modificado por
Perfil_aluno	Nome	Caracter	Usuário	Previamente	Usuário
	Id	Numérico	Mediador	Previamente	Mediador
	Idade	Numérico	Usuário	Previamente	Usuário
	Sexo	F M	Usuário	Previamente	Usuário
Perfil_aprendizagem	Estilo	Exemplos, Genérico-Específico, Princípios	Usuário	Previamente	Usuário
	Mídia_preferida	Som, gráfico, ling. natural	Usuário	Previamente	Usuário e mediador
	Tipo_exercício	Múltipla escolha, lacunas, subjetivo	Usuário	Previamente	Usuário e mediador
	Nível_escolaridade	2 grau, graduação e pós-graduação	Usuário	Previamente	Usuário
Perfil_motivacional	Característica_personalidade	Extrovertido introvertido	Usuário	Previamente	Usuário
	Motivação_grupo	Alta média baixa	Usuário	Dinâmico	Usuário e mediador
	Liderança	Líder não_líder	Usuário	Dinâmico	Usuário e mediador

	ansiedade	Alta média baixa	Usuário	Dinâmico	Usuário e mediador
Auto-avaliação	nome_tarefa	Numérico	Semiótico	Cada sessão	Semiótico
	Descrição tarefa	Caracter	semiótico	dinâmico	semiótico
	Confiança	Alta média baixa	Usuário e mediador	Dinâmico	Usuário e diagnóstico
	Competência	Alta média baixa	Usuário e mediador	Dinâmico	Usuário e diagnóstico
	Suporte	Nenhum parcial total	Usuário e mediador	Dinâmico	Usuário e diagnóstico
	Motivação	Alta média baixa	Usuário	Dinâmico	Usuário e mediador
	Internalização	0 1	Mediador	Dinâmico	Usuário e diagnóstico
	Estereótipo_tarefa	Novato intermediário especialista	Usuário	Dinâmico	Usuário e mediador
Diagnóstico_tarefa	Nome tarefa	Numérico	Semiótico	Cada sessão	Semiótico
	Habilidade_NDR ou Core	Lista de habilidades	Diagnóstico	Dinâmico	Diagnóstico
	Habilidade_ZDP	Lista de habilidades	Diagnóstico	Dinâmico	Diagnóstico
Suporte	Nome-tática	Caracter	Diagnóstico	Cada sessão	Diagnóstico
	Descrição	Caracter	Diagnóstico	Cada sessão	Diagnóstico
	Nível_suporte	Nenhum parcial total	Diagnóstico	Cada sessão	Usuário e Diagnóstico

Conforme observado na tabela 1, cada um destes componentes é constituído por um conjunto de atributos. O perfil_aluno corresponde a algumas características pessoais. O perfil de aprendizagem e motivacional contém as preferências de aprendizagem do aluno e algumas características motivacionais que devem ser consideradas no momento da realização das atividades.

O processo de diagnóstico inicia-se após o agente verificar se o aluno possui ou não modelo armazenado no sistema. Caso o aluno não possua, este deve preencher um cadastro e escolher o contexto no qual ele deseja trabalhar. Após esta etapa, o agente diagnóstico pode requisitar conteúdo pedagógico e um pré-teste ao agente semiótico (Jung, 2001), através do agente mediador (Quintans, 2002). O agente semiótico recebe a requisição do agente mediador sobre o tipo de conteúdo pedagógico que foi solicitado. Esse conteúdo é enviado para o mediador, para que ele disponibilize ao aluno, alguma tarefa e exercício associada a esta tarefa. Após realizar a tarefa, o aluno responde a sua auto-avaliação. A auto-avaliação além de auxiliar o aluno a analisar seu próprio desempenho, ela é usada como referência para que o agente diagnóstico avalie o diagnóstico das habilidades do aluno. Os atributos que compõem a auto-avaliação, o diagnóstico e o suporte são em sua maioria dinâmicos e serão atualizados na medida que o aluno interage com o sistema. A figura 1 apresenta a interface do framework para auto-avaliação do aluno. A modelagem e a implementação do processo de diagnóstico é discutida em detalhes na seção a seguir.

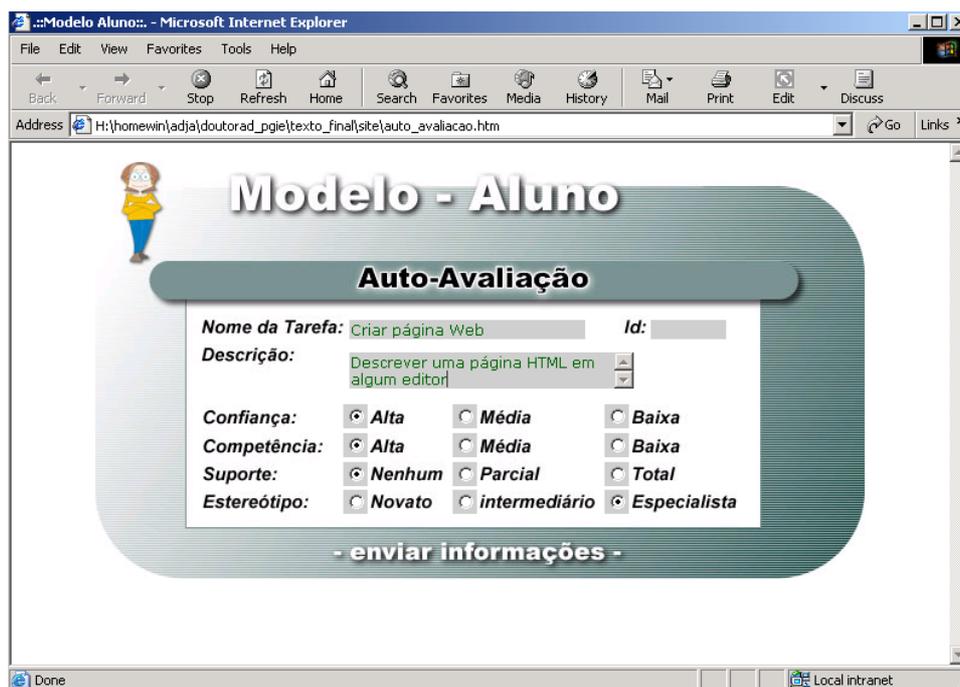


Figura 1: Auto-Avaliação

4. Modelagem e Implementação do Agente Diagnóstico segundo o Modelo BDI

Diversas abordagens têm emergido na literatura para modelagem de ambientes de aprendizagem orientada a agentes. A abordagem mentalista vê o agente como um sistema intencional, ou seja, tendo certos estados ou atitudes mentais geralmente atribuídas aos seres humanos. Seguindo esta abordagem mentalista, Bratman (1990) propõe o modelo BDI, representando as atitudes de crenças (**B**eliefs), desejos (**D**esires) e intenções (**I**ntentions).

Quando se tenta descrever o comportamento humano, é comum usar termos como “acredito”, “quero”, “desejo”, “necessito”. Estes termos são usados pelos seres humanos para explicar os comportamentos das pessoas baseados em estruturas (estados mentais) que supostamente formam a mente. BDI é, portanto, apenas uma das possíveis abordagens usadas atualmente para descrever o comportamento de agentes e suas funções. Ela permite descrever o agente diagnóstico em termos de suas crenças, desejos e intenções. A formalização do agente utilizando BDI permite descrever o processo de diagnóstico e raciocínio que o agente utiliza para detectar e dar suporte aos alunos que se encontram na zona de desenvolvimento proximal.

A facilidade de descrição e implementação, aliada a experiência do grupo de IA em implementar agentes usando tal arquitetura, foram alguns dos motivos que justificaram a escolha do BDI como abordagem para modelagem do agente diagnóstico. Para implementar o comportamento do agente, ou seja, as funcionalidades que dizem respeito ao papel de diagnóstico e suporte utiliza-se o modelo X-BDI (Mora, 1998). O X-BDI é uma ferramenta que permite a descrição formal de agentes cognitivos baseados no modelo BDI. Além da capacidade de especificação do agente através de estados mentais, o X-BDI possibilita a implementação destes agentes com alto nível de abstração, o que reduz a complexidade no desenvolvimento de sistemas baseado em agentes. O X-BDI permite que o projetista utilize o modelo BDI como uma ferramenta de descrição de agentes, mas também possibilita ao agente utilizar esta mesma descrição para representar os seus estados mentais e raciocinar sobre eles.

Esta seção mostra o funcionamento do agente diagnóstico através de uma coreografia executada na ferramenta X-BDI (Mora, 1998). A execução da coreografia inicia quando o agente identifica que há aluno conectado ao sistema (no exemplo corrente a aluna adja). Neste momento, o agente verifica se o aluno tem algum modelo já armazenado, caso tenha deve *carregar o modelo do aluno*. No exemplo, após tentar *ler a base de dados*, o agente identifica que não há modelo do aluno armazenado. Tentará então construir este modelo.

```

SICStus 3.0.3 (x86-win32-nt-4): Fri May 5 13:28:31 2000
File Edit Flags Settings Help

==> Kernel: Starting Choreography ...

==> Choreo: Receiving from Kernel ...

==> Choreo: Received! Press ENTER to continue
|:

==> Choreo: Sending to Kernel ...

current_time(2)
aluno_corrente(Adja,1)
==> Choreo: Press ENTER...
|:

==> Choreo: Receiving from Kernel ...

int_that(1,diagnostico,carrega_modelo(1),t_inf,[0.7])
int_to(diagnostico,act(diagnostico,le_base_dados(1)),t9)
int_to(diagnostico,menor(t9,t_inf),_46941)

==> Choreo: Received! Press ENTER to continue
|:

==> Choreo: Sending to Kernel ...

current_time(3)
contexto_corrente(internet)
act(2,act(diagnostico,le_base_dados(1)))
-(existe_modelo_armazenado(1))
==> Choreo: Press ENTER...
|:

```

Figura 2: Interface do agente diagnóstico verificando modelo do aluno

Ao observar a figura 2, pode-se notar que durante o processo de inicialização da Coreografia, o agente diagnóstico primeiramente testa se o aluno possui ou não modelo armazenado. Para realizar este teste, o agente executa o desejo *carrega_modelo*. Como não possui o modelo do aluno armazenado, passará a estar ativo o desejo do agente *constroi_modelo*. Detalhes da especificação ou formalização do agente usando os estados mentais *crença* (*bel*) e desejos (*des*) podem ser vistos na figura 3 a seguir. Conforme foi mencionado anteriormente na seção 3, para construir o modelo do aluno, o agente diagnóstico deve antes receber do aluno a informação do contexto no qual o aluno deseja trabalhar. Após receber esta informação, o agente diagnóstico deve *fazer uma solicitação ao agente semiótico das tarefas, habilidades e conceitos relacionadas a tarefa*.

```

des(diagnostico, carrega_modelo(Id), Tf, [0.7])
    if    bel(diagnostico, aluno_corrente(Aluno, Id)),
        bel(diagnostico, existe_modelo_armazenado(Id)).
act(diagnostico, le_base_dados(Id))
    causes    bel(diagnostico, carrega_modelo(Id)).
des(diagnostico, constroi_modelo(Id, Tarefa), Tf, [0.7])
    if    bel(diagnostico, aluno_corrente(Aluno, Id)),
        bel(diagnostico, tarefa(Tarefa)),
        bel(diagnostico, -existe_modelo_armazenado(Id)),
        -bel(diagnostico, perfil_cognitivo_aprendiz(Id,
Perfil)).
act(diagnostico, envia_tarefa_pre_teste(Id, Tarefa))
    causes    bel(diagnostico, constroi_modelo(Id, Tarefa)).

```

Figura 3: Estados Mentais do Agente Diagnóstico

Após receber a descrição da tarefa e suas habilidades, o agente deve executar a ação *envia_tarefa_pre_teste* para iniciar a construção do modelo. O pré-teste tem como objetivo a análise do estado corrente do conhecimento do aluno, ainda sem qualquer suporte. Após a realização do pré-teste, o agente diagnóstico terá condições de avaliar o desempenho do aluno, suas dificuldades, sua

confiança, sua necessidade de suporte. Alguns destes parâmetros são extraídos de uma auto-avaliação do próprio aluno; outros, são sensorados de seu desempenho ao realizar a tarefa. Depois de receber todas as informações sobre a realização do pré-teste do aluno e avaliar seu grau de competência, confiança e suporte na realização da tarefa, o agente diagnóstico é capaz de detectar quais as habilidades do aluno se encontram no Core e quais se encontram na ZDP, conforme pode ser visualizado na execução da consulta ilustrada na figura 4.

```

SICStus 3.8.3 (x86-win32-nt-4): Fri May 5 13:28:31 2000
File Edit Flags Settings Help
confianca_auto(colocar_fundo,1,alto)
competencia_ext(colocar_fundo,1,sucesso)
suporte_ext(colocar_fundo,1,sup_nenhum)
confianca_auto(inserir_figura,1,alto)
competencia_ext(inserir_figura,1,sucesso)
suporte_ext(inserir_figura,1,sup_parcial)
confianca_auto(inserir_texto,1,alto)
competencia_ext(inserir_texto,1,sucesso)
suporte_ext(inserir_texto,1,sup_nenhum)
confianca_auto(inserir_hiperlink_web,1,alto)
competencia_ext(inserir_hiperlink_web,1,insucesso)
suporte_ext(inserir_hiperlink_web,1,sup_parcial)
confianca_auto(inserir_hiperlink_email,1,alto)
competencia_ext(inserir_hiperlink_email,1,insucesso)
suporte_ext(inserir_hiperlink_email,1,sup_nenhum)
==> Choreo: Press ENTER...
|:

==> Choreo: Receiving from Kernel ...

==> Choreo: Received! Press ENTER to continue
|:

yes
| ?- setof(T,
demo(holds_at(bel(diagnostico, na_zdp(T,001)), 6)),
ZDP).

ZDP = [construir_pagina,inserir_figura,inserir_hiperlink_email,inserir_hiperlink_web] ?

yes
| ?- setof(T,
demo(holds_at(bel(diagnostico, no_core(T,001)), 6)),
CORE).

CORE = [colocar_fundo,construir_pagina,inserir_texto] ?

yes
| ?-

```

Figura 4: Diagnóstico Identificando Habilidades na ZDP e no Core

Após detectar as habilidades do aluno que se encontram na ZDP, inicia-se o processo de suporte propriamente dito, onde o agente diagnóstico irá verificar no banco de táticas existentes qual a tática adequada ao aluno. Para que a tática seja enviada em consonância com o nível de dificuldade do aluno, o agente diagnóstico deve consultar informações sobre a competência do aluno ao realizar a tarefa (informações enviadas pelo agente mediador), além da própria auto-avaliação realizada pelo aluno.

Para a escolha do nível de suporte, utilizou-se a proposta de Kay (2001) baseada na noção de estereótipos. O aluno é classificado como *novato*, *intermediário* e *expert (especialista)*. O uso de estereótipos não tem como objetivo estabelecer rótulos ou discriminar os estudantes. Ele está sendo usado apenas como uma forma de estabelecer o modelo inicial do aprendiz antes de aplicar a tática de suporte ao aluno. É importante notar que o uso de estereótipos não é algo estático. O aluno pode ser em um dado momento novato em alguma tarefa e posteriormente um especialista em outra.

Em consonância com o estereótipo ou nível do aluno, podem ser aplicados diferentes níveis de suporte: parcial ou total. O suporte total é um nível avançado de suporte, adequado para os alunos que necessitam de máxima assistência e que encontram muita dificuldade na realização da habilidade. O suporte parcial é um nível moderado geralmente pode ser usado para o aluno que tem o estereótipo intermediário e requer ajuda parcial em alguma habilidade. Este nível é indicado para os alunos que têm um alto grau de confiança e competência na execução da habilidade e necessitam apenas de alguma “dica” para obter êxito ou sucesso na tarefa. A escolha por níveis de suporte foi inspirada no trabalho ECOLAB (Luckin e Boulay, 1999a), onde os autores descrevem cinco diferentes níveis possíveis de ajuda. O suporte inicia em um nível mais baixo e vai sendo atualizado para níveis mais alto de suporte caso haja necessidade.

5. Considerações Finais

Este trabalho ampliou o conjunto de resultados no que concerne ao modelo de aluno e apontou diversas potencialidades da arquitetura BDI enquanto um modelo de especificação de agentes e ao mesmo tempo enquanto uma ferramenta. A aplicação da ferramenta XBBDI se mostrou bastante promissora para a modelagem de agentes pedagógicos. Desta forma, ampliou-se os resultados previamente atingidos em trabalhos anteriores desenvolvidos por Giraffa (1999) e Bercht (2000).

Uma das principais contribuições deste trabalho foi aplicar pressupostos da teoria de Vygotsky para o projeto de um ambiente de aprendizagem computacional. O modelo de aluno foi concebido de forma interdisciplinar, em que o conceito de ZDP e táticas pedagógicas de suporte foram modeladas usando uma arquitetura BDI, baseada em crenças, desejos e intenções. Este artigo buscou determinar como construir um modelo do aluno e quais informações deve conter o modelo. A tarefa de mediação ou suporte dentro da zona de desenvolvimento proximal permitiu identificar o potencial da modelagem para formação de grupos, no qual um indivíduo que possui uma habilidade na ZDP possa interagir com outro especialista que possui habilidade internalizada, ou seja, no *Core*.

A formalização do agente diagnóstico resultou em uma representação das noções de Domínio (com suas atividades, habilidades, conceitos), do *Core* e ZDP. O modelo de aluno foi inicializado através da interação com o próprio usuário, sua auto-avaliação e a noção de estereótipos.

Como trabalhos futuros sugere-se estender o modelo de aluno para um modelo de grupo efetivamente construído. A partir deste momento, o agente diagnóstico estará apto a detectar ZDP não apenas individuais, como também, ZDP_coletivas.

Referências

- Andrade, A. F. (2003). Uma Aplicação da Abordagem Sociointeracionista de Vygotsky para a Construção de um Ambiente Computacional de Aprendizagem. Porto Alegre: UFRGS, 2003. Tese (Doutorado) – Programa de Pós Graduação em Informática na Educação- PGIE -, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2003.
- Andrade, A. F.; Brna, P.; Vicari, R. M. (2002a). A Diagnostic Agent based on ZPD approach to improve Group Learning. In: ITS 2002. Workshop on Individual and group modelling methods that help learners understand themselves. Intelligent Tutoring Systems Conference. San Sebastian, Espanha. Junho 2 – 4, 2002.
- Andrade, A. F.; Brna, P.; Vicari, R. M.; Mora, M. (2002b). Cognitive Diagnosis Based on ZPD and Core Approaches. In: International Conference on Artificial Intelligence - IC-AI'02, 2002, Las Vegas. Proceedings of the International Conference on Artificial Intelligence- IC-AI'02. Junho 24 – 27, 2002.
- Andrade, F. A. ; Jaques, P, A; Vicari, M. R.; Bordini, R. H; Jung, L. J. (2001). A Computational Model of Distance Learning Based on Vygotsky's Socio-Cultural Approach. Workshop of Multi-Agent Architectures for Distributed Learning Environments. Proceedings of AIED 2001- X International Conference on Artificial Intelligence on Education, San Antonio, Texas, 19-23, May, 2001.
- Andrade, F. A. ; Jaques, P, A; Vicari, M. R.; Bordini, R. H; Jung, L. J. (2000). Uma Proposta de Modelo Computacional de Aprendizagem à Distância Baseada na Concepção Socio-Interacionista de Vygotsky. II Workshop de Ambientes de Aprendizagem Baseado em Agentes. X Simpósio Brasileiro de Informática na Educação – SBIE, Maceió, Alagoas, novembro de 2000.
- Bercht, M; Vicari. M. R. (2000). Pedagogical agents with affective and cognitive dimensions. In: RIBIE, Vina del Mar, 2000.
- Bratman, M. E. (1990). What Is Intention? In: Cohen, P; Morgan, J; Pollack, M. (Eds.). Intentions in communication; MIT Press, Cambridge, MA, 1990.
- Bull, S. and Smith, M. (1997). A pair of Student Models to Encourage Collaboration. User Modelling: Proceedings of the sixth International Conference, UM97. Vienna, New York: Springer Wien New York, 1997.
- Bull, S.; Pain, H. (1995). “Did I say what I think I said, and do you agree with me?”: Inspecting and questioning the student model. In Proceeding of 7th World Conference on Artificial Intelligence in Education, AACE, 501-508, 1995.

- Giraffa, L. M. (1999). Uma arquitetura de tutor utilizando estados mentais. Tese de Doutorado em Ciência da Computação – Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, 1999.
- Jaques, P. A.; Bercht, M.; Bocca, E.; Vicari, M. R. (2003). Cognitive Reasoning to Respond Affectively to the Student. In: The IASTED International Conference on Computers and Advanced Technology in Education. Proceedings. June 30 - July 2, 2003. Rhodes, Greece.
- Jung, J. L.; Jaques, P. A.; Andrade, A. F.; Bordini, R.; Vicari, M.R. (2001). Um Agente Inteligente Baseado na Engenharia Semiótica Inserido em um Ambiente de Aprendizado à Distância. In: Workshop Sobre Fatores Humanos em Sistemas Computacionais, IHC, 4., 2001, Florianópolis, SC. Anais... Florianópolis: UFSC, 2001.
- Kay, J. (2001). User Modelling for adaptation. In User Interfaces for all: concepts, methods and tools. Constantine Stephanides (Ed.), Lawrence Erlbaum Associates, Inc, EUA, 271-293, 2001.
- Lewis, R. (2000). Human activity in learning societies. Invited Paper. In: proceeding of ICCE/ICAI, Taipei, Taiwan, 2000.
- Luckin, R.; Boulay, B. D. (1999a). Ecolab: The Development and Evaluation of a Vygotkian Design Framework. International Journal of Artificial Intelligence in Education, 10, 198-220, 1999.
- Luckin, R.; Boulay, B. D. (1999b). Capability, Potential and Collaborative Assistance. In: proceeding of Seventh International Conference in User Modelling, UM 99. Springer Wien, New York, 139-148, 1999.
- Luckin, R. (1996). TRIVAR: Exploring the “zone of proximal development”. In: Proceedings of European Conference on Artificial Intelligence. P. Brna, A. Paiva, J. Self (Eds.). Colibri Press, Lisbon. 16-22, 1996.
- Mora, M.C.; Vicari, R.M; Lopes, J.G.P; Coelho, H. (1998). BDI Models and systems: reducing the gap . ATAL'98 –Paris, July, 1998.
- Quintans, K; Pilatti, L; Santos, L. (2002). Agente de Mediação entre Agentes Artificiais e Humanos em um sistema multiagente de apoio a aprendizagem a distância. Trabalho de conclusão de curso, FACIN/PUCRS, 2002.
- Self, J. A. (1988). Student Models: what use are they? In: Artificial Intelligence Tools in Education. Ercolli and Lewis (eds), Elsevier Science Publishers B. V., North- Holland, IFIP, 1988.
- Vygotsky, L. S. (1998). A Formação Social da Mente: o Desenvolvimento dos Processos Psicológicos Superiores. Editora Martins Fontes, São Paulo, 1998.
- Weiss, G.(2000). Multiagent Systems: a modern approach to distributed artificial intelligence. Edited by Gerhard Weiss. The MIT Press: Cambridge, Massachusetts, London, England, 2000.