



KURRUPAKO: UM AGENTE ANIMADO SÓCIO-AFETIVO PARA AMBIENTES DE APRENDIZAGEM

Eliseo Berni Reategui¹, Elisa Boff^{1,2}, Rafael Fernando Ceron¹

¹ Departamento de Informática – Universidade de Caxias do Sul (UCS). Rua Francisco Getúlio Vargas, 1130 – 95070-560 Caxias do Sul – RS – Brasil.
{ebreateg, eboff, rfceron}@ucs.br

Rosa Maria Vicari²

² Instituto de Informática – UFRGS - Bloco IV, Av. Bento Gonçalves, 9500 – Porto Alegre – RS
rosa@inf.ufrgs.br

Resumo. Este artigo detalha a modelagem e implementação de um agente de interface sócio-afetivo capaz de interagir com alunos de um ambiente de aprendizagem de algoritmos. O agente monitora a interação com o usuário e utiliza as informações coletadas para fazer recomendações personalizadas de conteúdo. Além disso, o agente é capaz de identificar outros estudantes conectados no sistema que podem servir como tutores em determinados tópicos, e sugere a interação entre estes e alunos com dificuldades. Através destas recomendações, o agente motiva a formação de grupos e promove a aprendizagem colaborativa.

Palavras-chaves: Agentes Animados de Interface, Ambientes de Aprendizagem, Sistemas de recomendação, Afetividade, Interação Social e Formação de Grupos.

KURRUPAKO: A SOCIAL-AFFECTIVE ANIMATED AGENT FOR LEARNING ENVIRONMENTS

Abstract. This paper describes a social-affective animated agent that is able to interact with students in a learning environment for algorithms. The agent monitors the interaction with the user and employs the information collected to make personalized recommendations of contents. In addition, the agent is capable of identifying other students that may act as tutors in certain topics, and suggests the interaction between them and the students that are having difficulties in their studies. Through these recommendations, the agent motivates group formation and promotes collaborative learning.

Keywords: Interface Agents, Learning Environments, Recommender Systems, Affective, Social Interaction and Workgroups.

1. Introdução

Ao longo dos anos, vários pesquisadores, tanto da Psicologia quanto da Educação, têm levantado evidências acerca da importância da socialização na construção do conhecimento. Neste grupo podemos incluir Vygostky (1984) e Maturana (1995). Piaget (1976) também não desconheceu a importância das trocas sociais para o desenvolvimento cognitivo, apesar de tê-la abstraído na grande maioria dos seus estudos (Costa, 2003). Esse autor aponta o conhecimento humano como sendo essencialmente coletivo e diz que a vida social constitui um dos fatores essenciais da formação e do crescimento dos conhecimentos pré-científicos e científicos (Costa, 2003).

As trocas sociais são enfatizadas no processo de construção do conhecimento, mesmo sabendo-se que a aprendizagem ocorre individualmente. Estas atividades coletivas de



aprendizagem normalmente são divididas em aprendizagem cooperativa ou colaborativa, existindo diferentes conceitos para estes termos. Na literatura, podem-se encontrar autores que consideram ambos com o mesmo significado, outros que diferenciam tais termos. A aprendizagem colaborativa¹ ocorre como resultado do compartilhamento do conhecimento entre múltiplos participantes. Para tanto, aparece a tecnologia computacional que apóia e organiza as situações, facilitando este processo. Um processo de aprendizagem nesses moldes, baseado na coletividade, difere-se dos tradicionais pelo fato de apresentarem ausência de hierarquia formal, possuírem um objetivo em comum, respeitarem às diferenças individuais e possibilitarem a liberdade de expressão na exposição de idéias e questionamentos.

Pelas idéias expostas acima, podemos afirmar que se a socialização é importante no contexto da aprendizagem, também o é em ambientes informatizados que visem a facilitação deste processo, tais como os Sistemas Tutores Inteligentes (STI) e os Ambientes Inteligentes de Aprendizagem (AIA). Observa-se nos principais meios de divulgação de avanços científicos, uma crescente utilização de técnicas de Inteligência Artificial para adequar sistemas computacionais às necessidades individuais dos usuários, ou seja, na individualização do processo de aprendizagem.

Baseado nos conceitos já apresentados e ressaltando a idéia de que a interação social é fator importante para o desenvolvimento do conhecimento, este trabalho propõe um agente animado sócio-afetivo, que a partir do perfil individual e do grupo, recomenda conteúdos de algoritmos e sugere a interação entre os alunos, a fim de que os próprios alunos assumam o papel de tutores.

O objetivo é tentar amenizar as dificuldades pedagógicas que possam acontecer com algum estudante no decorrer de uma sessão de ensino, através da recomendação de alunos potencialmente capacitados e amigáveis e, a partir daí, proceder a formação de grupo entre o aluno com dificuldades e o “colega tutor”. À medida que esta dinâmica vai se estabelecendo, novos grupos vão se formando. Com o passar do tempo, as recomendações do agente animado se tornam mais adequadas, visto que o perfil do aluno consultado para gerá-las vai sendo atualizado a cada interação com o sistema.

Este artigo está organizado da seguinte forma: a seção 2 aborda os aspectos teóricos que influenciaram a modelagem do agente animado proposto, a seção 3 apresenta o modelo para o agente animado e o Ambiente para Aprendizagem de Algoritmos 3A, nosso estudo de caso. Na seção 4 são apresentadas as considerações finais e possibilidades para trabalhos futuros.

2. Trabalhos Relacionados

A área de agentes de interface é voltada para o desenvolvimento de aplicações que visam aperfeiçoar o processo de interação entre humanos e computadores através da utilização de personagens que podem ser animados ou não.

Cosmo (Lester, 1997) e Adele (Shaw, 1999) são alguns exemplos clássicos desses agentes. Cosmo é um agente voltado para área da informática, mais especificamente para área de redes no roteamento de pacotes no domínio da Internet. Em tempo real, ele procura demonstrar e aconselhar a melhor forma de enviar pacotes para um determinado destino. Adele, por sua vez, é um agente com características humanas projetado para

¹ Usamos o conceito de “colaboração” como o trabalho conjunto das partes envolvidas, não envolvendo a interação no trabalho individual de cada um. As contribuições individuais não são alteradas por outros participantes do grupo de trabalho, enquanto que o trabalho cooperativo apresenta a característica de interação e trabalho conjunto nas partes individualmente realizadas.



trabalhar com os estudantes na área da medicina. Suas funções são auxiliar na resolução de problemas e disponibilizar material para estudo. Além disso, Adele é capaz de destacar assuntos importantes, monitorar o aluno e realizar testes de avaliação com o objetivo de verificar o grau de compreensão do aluno sobre o assunto que está sendo estudado.

Outros pesquisadores têm investigado o impacto destes agentes em ambientes de aprendizagem interativo. André et al. (1999), por exemplo, demonstraram que estudantes podem considerar o assunto em estudo significativamente menos difícil e a apresentação mais lúdica na presença de um assistente virtual. Neste mesmo experimento a maior parte dos estudantes afirmou que o assistente os auxiliou a prestar atenção a informações relevantes. Já Craig et al. (2002) mostraram efeitos distintos no aprendizado de alunos submetidos a interações com agentes estáticos e animados.

Outros pesquisadores focalizam a importância de aspectos sociais e afetivos nas experiências de aprendizagem dos estudantes em ambientes computacionais interativos. Elliot et al. (1999) investigaram a importância da dimensão afetiva em personagens virtuais, desenvolvendo estudos sobre agentes pedagógicos que são sensíveis aos estados emocionais dos estudantes e são capazes de utilizar dados aspectos afetivos na resolução de problemas. De Angeli et al. (2001) também apresentaram argumentos que dão suporte à utilização de personagens virtuais, enfatizando que, através da introdução de estímulos sociais, os agentes de interface podem melhorar a comunicação entre o usuário e o computador.

Conati (2002) desenvolveu um modelo probabilístico para monitor as emoções dos usuários e seu entrosamento na interação com um jogo educacional que objetiva a aprendizagem da fatoração de números. O modelo avalia a afetividade através da evidência das possíveis causas da manifestação de emoção do usuário e de seus efeitos (expressões corporais influenciadas por reações emocionais). O modelo probabilístico desenvolvido é usado na decisão do agente pedagógico para gerar intervenções visando alcançar o melhor entre a aprendizagem do aluno e seu engajamento durante sua interação com o jogo. O agente pedagógico do jogo “*Prime Climb*” foi desenvolvido com os recursos do *Microsoft Agent*, assim como o agente proposto neste trabalho descrito na seção 3.

A Educação é uma área onde os agentes de interface podem ser aplicados como um recurso interessante para aliviar a carga cognitiva do aluno, tanto em relação à navegação na interface, como no acesso do conteúdo desejado. Neste caso, o agente atua como um tutor virtual que guia o aluno pelos ambientes interativos, como em (Shaw, 1999), onde instrutores virtuais guiam os estudantes em histórias interativas *online*.

Na pesquisa realizada por Ball et al. (1997) são elencadas as características de interfaces sociáveis, a saber:

- Suporte à interatividade através de perguntas e respostas: Os agentes não somente podem responder quando lhes é perguntado alguma coisa, mas também realizam perguntas para melhor compreender sua tarefa, descrever seus planos e antecipar problemas;
- Reconhecimento dos custos de interação e atraso: o agente deve modelar a significância de suas decisões e o potencial custo de um possível erro. Deste modo, ele só buscará uma interação como o usuário quando realmente necessário, não o desconcentrando, nem atrasando uma tarefa à espera de uma resposta caso o usuário esteja ausente;



- Gerenciamento de interrupções: o agente deve ser capaz de perceber se o usuário está ativo ou inativo no ambiente para escolher o momento apropriado de realizar a interrupção;
- Reconhecimento dos aspectos sociais e emocionais da interação: um agente de interface deve saber qual o “comportamento adequado” para executar determinada tarefa. Também deve saber quando dar bom dia ou boa tarde, e conhecer o tipo de humor a apresentar para melhorar sua interação com o usuário. São estas características que proporcionam boa parte da vivacidade de uma interface.

Normalmente ambientes de aprendizagem são sistemas de múltiplos usuários e, portanto, os agentes devem adaptar-se à forma de estudo de cada um. A próxima seção apresenta nosso experimento com o agente de interface *Kurrupako* em um ambiente multi-usuário para aprendizagem de algoritmos.

3. O agente animado sócio-afetivo

O 3A é um ambiente virtual de aprendizagem utilizado simultaneamente por vários usuários, onde o agente proposto é capaz de aprender e se adaptar a cada estudante. Este agente foi batizado de *Kurrupako* e possui como uma de suas principais funcionalidades a capacidade de recomendar conteúdos e tutores aos usuários de forma animada.

3.1 Ambiente de Aprendizagem de Algoritmos – 3A

O ambiente interativo de aprendizagem de algoritmos 3A vem sendo desenvolvido como forma de apoio à disciplina de algoritmos do Departamento de Informática da Universidade de Caxias do Sul, uma disciplina basicamente presencial. O principal objetivo do 3A é tornar as aulas mais dinâmicas, aumentando o interesse e participação dos alunos e provendo um ambiente onde eles possam resolver dúvidas na ausência do professor. O sistema apresenta aos alunos conteúdos regulares da disciplina, propõe exercícios, disponibiliza um mecanismo de interação síncrona além de uma ferramenta para teste e execução de algoritmos.

O 3A é um sistema desenvolvido para Web utilizando-se da linguagem Java (J2SE²), que além de ser uma linguagem robusta, provê meios de implementação para a Internet através de recursos da plataforma J2EE (Java Enterprise Edition³). Além disso, utiliza o Tomcat⁴ como servidor para que esse ambiente possa ser iniciado e acessado a partir de um navegador. Como banco de dados foi escolhido o MySQL⁵, por ser um banco leve e de simples utilização. Basicamente, pode-se dizer que o 3A é constituído por quatro componentes, descritos nas subseções a seguir:

3.1.1 Módulo de Conteúdo

O Módulo de Conteúdo é responsável pela organização e armazenamento dos conteúdos e atividades do sistema. É constituído basicamente de uma apostila de algoritmos e de exercícios cadastrados para cada tópico por nível de dificuldade.

3.1.2 Módulo para Recomendações de conteúdo

O módulo de recomendação de conteúdo é utilizado para sugerir um *link*, um exercício ou um material relacionado ao assunto consultado pelo aluno. Um sistema de

² <http://java.sun.com/j2se/index.jsp>

³ <http://java.sun.com/j2ee/index.jsp>

⁴ <http://tomcat.apache.org>

⁵ <http://www.mysql.com>



recomendação é empregado para identificar os itens a serem sugeridos a este aluno a partir da análise das seqüências de navegação executadas pelos estudantes (Reategui, 2005). Para viabilizar este processo, um sistema de monitoramento armazena todas as interações dos alunos, indicando o dia e a hora em que cada seção sistema é acessada por cada um. Um sistema de mineração de dados vasculha este repositório em busca de seqüências de acesso freqüentes. Quando encontradas, estas seqüências são armazenadas em descritores de itens (Reategui, 2004). Estes descritores empregam parâmetros como fator de confiança (FConf) e fator de suporte (FSup) para indicar a força dos relacionamentos entre itens:

- *FConf* constitui-se na razão do número de usuários que acessou tanto o item I_1 quanto I_2 , pelo número de usuários que acessou apenas o item I_1 . Note que o acesso a um item representa aqui a consulta a um texto ou a realização de uma atividade.

$$FConf = |I_1 \cap I_2| / |I_1|$$

- *FSup* constitui-se na razão do número de usuários que acessou tanto o item I_1 quanto I_2 , sobre o número total de usuários (n).

$$FSup = |X \cap Y| / n$$

Através deste método, podemos identificar, por exemplo, que alunos que consultam o tópico “Recursão” também consultam o tópico “Chamadas de funções”, e implementam alguns exercícios sobre recursão.

3.1.3 Módulo Recomendação de tutores

Este módulo tem como objetivo encontrar alunos que possam auxiliar outros na resolução das tarefas propostas, estimulando a cooperação e a aprendizagem colaborativa. A função de recomendação de tutores tem como parâmetros de entrada o identificador do aluno e o tópico no qual se necessita de ajuda. Os alunos conectados no sistema são identificados e cria-se um escore individualizado para cada um. Este escore é baseado nos seguintes aspectos:

- **Estado de humor:** O humor de uma pessoa pode variar diversas vezes em um dia, dependendo do conjunto de ações que esta realiza. Mesmo tendo sucesso na maior parte de suas atividades, o humor da pessoa pode ser afetado negativamente, deixando-a com uma espécie de resíduo emocional desagradável. Aqui, utilizamos o estado de humor como indício da capacidade do estudante em ser tutor em determinado momento, já que um aluno num mau estado de humor pode ter mais dificuldade em auxiliar seus colegas. Atualmente, o estado de humor não é inferido pelo sistema, mas é informado espontaneamente pelo estudante no momento do *login* através de uma interface gráfica. A variável pode assumir valores no intervalo $[0,1]$, onde 0 representa mau humor e 1 representa ótimo humor.
- **Performance:** Representa o desempenho do aluno nas diversas atividades desenvolvidas. Um maior número de exercícios concluídos e um maior número de acertos denotam uma performance melhor. Utilizamos dois graus de performance relativos para estimar o desempenho do aluno. O primeiro mede o quanto o aluno apresentou uma boa performance em seu percurso durante o semestre/ano. O outro estabelece o quanto o aluno conseguiu aprender sobre o tópico específico no qual está sendo solicitada a ajuda. Ambos os graus são calculados pela mesma fórmula, uma delas aplicada a todas as leituras e atividades propostas no semestre/ano, e a outra aplicada às leituras e atividades propostas para o tópico em questão.

$$Performance = \frac{P1 * Leituras_concluídas}{Leituras_propostas} + \frac{P2 * Atividades_concluídas_com_êxito}{Atividades_propostas}$$



$P1$ e $P2$ são pesos que permitem ponderar a importância de cada fator. Deste modo, é possível por exemplo atribuir um peso maior para o fator *Atividades concluídas com êxito*, já que o que se busca é verificar o desempenho do aluno. O grau de performance final é calculado através da média entre os dois graus relativos.

- **Aceitação:** No final de cada sessão em que um aluno atuou como tutor, uma nota lhe é atribuída pelo colega que solicitou o auxílio, indicando o quão proveitosa foi a interação. O grau de Aceitação é calculado pela média destas notas, indicando o quanto o aluno é aceito como tutor por seus colegas.
- **Média Interações:** Refere-se à média total de comunicações iniciadas e respondidas por um aluno, e pode ser calculada pela expressão:

$$MédiaInterações = \frac{P1 * Com_iniciadas}{MGeral_com_iniciadas} + \frac{P2 * Com_respondidas}{MGeral_com_respondidas}$$

onde *Com_iniciadas* representa o número de comunicações iniciadas pelo aluno, *MGeral_com_iniciadas* a média geral de comunicações iniciadas pelos alunos, *Com_respondidas* é o número de comunicações respondidas pelo aluno e *MGeral_com_respondidas* é a média geral de comunicações respondidas pelos alunos, enquanto $P1$ e $P2$ são pesos atribuídos para ponderar a importância de cada fator.

O escore final do aluno é obtido através de uma fórmula simples que calcula o somatório de todos os dados considerados para a recomendação de tutores.

$$EscoreAluno = P1 * Performance + P2 * EstadoHumor + P3 * Aceitação + P4 * MédiaInterações$$

No caso específico da recomendação de tutores, apenas para efeito de teste e sem a pretensão de levantar questionamentos sobre a importância de cada um destes fatores no processo de recomendação, todos os fatores são multiplicados pelo mesmo peso com valor 1.

3.1.4 O Agente de Interface

O Agente de Interface tem o principal objetivo de apresentar aos usuários as recomendações de conteúdo e as sugestões de alunos tutores geradas pelo sistema. *Kurrapako* foi desenvolvido de forma a proporcionar aos estudantes um meio mais amigável de interação, uma vez que o agente é capaz de tomar iniciativas e executar determinadas tarefas sem que isso tenha sido solicitado.

Pelo fato de reproduzir animações, gestos e representações conhecidas, salientando bem suas características sociais, o agente ganha maior aceitação e maior grau credibilidade por parte do usuário, gerando nele maior confiabilidade. *Kurrapako* realiza quatro tarefas bem definidas, ilustradas na Figura 1:

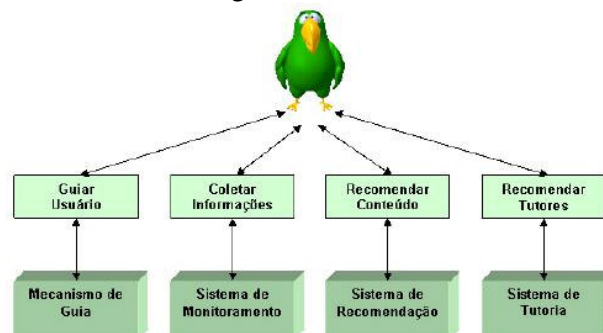


FIGURA 1 – Agente Kurrupako

A recomendação de conteúdo e de alunos tutores é feita através do acesso aos módulos apropriados, descritos nas seções anteriores (3.1.2 e 3.1.3).

Já o mecanismo de guia do usuário é capaz de conduzi-lo pelo ambiente, apresentando-lhe a organização do conteúdo, as ferramentas disponíveis, bem como a possibilidade e a importância de interação com os colegas na realização das atividades. Além disso, o mecanismo de guia do usuário utiliza o estado afetivo do aluno (ex. feliz, triste) para tentar atingi-lo psicologicamente com alguma frase que o faça refletir sobre determinada situação. Estas frases estão armazenadas numa tabela no banco de dados e são escolhidas de acordo com o estado afetivo do aluno. Desta forma, pretende-se motivar o aluno e auxiliá-lo a ultrapassar barreiras relacionadas a dificuldades encontradas em determinado assunto, ou mesmo relacionadas ao seu estado afetivo.

O módulo do agente responsável pela coleta de informações é por sua vez responsável pelo monitoramento de todas as ações do usuário, fornecendo os subsídios necessários para que o agente possa tratar cada aluno de maneira personalizada. *Kurrupako* foi implementado a partir dos modelos de Agentes da Microsoft⁶. Esses agentes são componentes prontos que contêm uma série de recursos como: animações, expressão através da fala, expressões faciais e gestos. O agente *Kurrupako* teve suas ações programadas através de scripts em linguagem VBScript e inseridas em páginas HTML. Para criar as ações do agente foi desenvolvido um mecanismo de geração e interpretação dinâmica de scripts. Como o A3 foi construído através de páginas JSP, que por sua vez geram como saída código HTML, a idéia principal do projeto de implementação do agente foi utilizar a potencialidade de programação em JSP e Java para gerar scripts dinâmicos.

acoesAgente	estadoAfetivo	tipoAcao	fala	movimento
1	A	F	Não esqueça de mim quando precisar de ajuda!	LookUpLeftBlink
2	A	F	Olá, que bom vê-lo novamente!!!	GetAttention
3	T	F	O que eu posso fazer para lhe deixar feliz?	Thinking
4	NA	T	Por que você não conversa com	LookLeftReturn
5	NA	T	Um colega seu poderia lhe ajudar:	Uncertain
6	NA	R	Gostaria de lhe recomendar um exercício:	LookRightReturn

FIGURA 2 – Tabela de Animações

O agente possui um Módulo de Animação que seleciona uma animação apropriada para a tarefa que será executada. Para este módulo foi criada uma tabela no banco de dados representada na Figura 2. A coluna da tabela representando **estadoAfetivo** é utilizada para estabelecer o tipo de fala e de animação que mais se enquadra ao estado afetivo do aluno: A (alegre) ou T (triste). A coluna **tipoAcao** indica o tipo de tarefa que o agente pode realizar: F (Guiar), R (Recomendar Conteúdo), T (Recomendar Tutor), N (Fazer Nada – útil para não interromper o usuário a todo o momento). A coluna **fala** contém a frase a ser utilizada pelo agente quando executa determinada tarefa. Por fim, a coluna **movimento** armazena o identificador da animação a ser executada para a tarefa. A Figura 3 apresenta o agente Kurrupako recomendando alguns conteúdos para o estudante Rafael durante a interação do usuário no ambiente A3.

⁶ <http://www.microsoft.com/msagent>
V. 4 Nº 1, Julho, 2006



Seja bem vindo(a),
Rafael Cerón

Logout

Apostila

- 1. Introdução
 - 1.1 Exemplo 1.1
 - 1.2 Exemplo 1.2
- 2. Expressões
- 3. Variáveis
 - 3.1 Exemplo 3.2
 - 3.2 Exemplo 3.3
 - 3.3 Exemplo 3.11
- 4. Estrutura Sequencial
 - 4.1 Exemplo 4.2
 - 4.2 Exemplo 4.3
- 5. Estrutura Condicional
 - 5.1 Exemplo 5.7
 - 5.2 Exemplo 5.8
 - 5.3 Exemplo 5.9

6.2 Comando para ... faça

O comando para ... faça também permite a descrição, dentro de um algoritmo, de uma de repetição. Sua forma geral é:

```
para <variável de controle> de <valor inicial> ate <valor final> [passo
<incremento>] faça
<lista de comandos>
```

fimpara

Na estrutura para ... faça, a variável de controle é inicializada com e no início de cada iteração, seu valor é comparado com . Se o valor da variável for menor ou igual a , a lista de comandos é executada e após ser executado o último comando da lista, a variável de controle é incrementada. Isto repete-se até que o valor da variável de controle seja maior que , quando então é executado o comando imediatamente após a palavra fimpara. A instrução passo é necessária se o incremento for diferente de 1.

Rafael tenho algumas recomendações para você!
Sugiro estudar: Formato de algoritmos que possuem estrutura de repetição, Cálculo da duração em horas e minutos de um jogo.

FIGURA 3 – Recomendação de Conteúdo

Para realizar essa recomendação, todos os módulos da arquitetura proposta foram acionados: (1) o usuário foi identificado; (2) seu histórico de leituras e atividades propostas foi recuperado; (3) uma das ações do agente foi selecionada, no caso *recomendar conteúdo*; (4) após a seleção da ação de recomendação de conteúdo, os itens relacionados ao assunto consultado pelo aluno foram encontrados pelo sistema de recomendação; (5) a fala do agente Kurrupako foi então construída pelo gerador dinâmico de scripts com base nos dados da tabela de controle das ações do agente e nas recomendações advindas do sistema de recomendação; (6) as animações e falas do agente Kurrupako foram então acionadas para gerar a visualização apropriada. A Figura 4 mostra outro exemplo de interação com o agente Kurrupako em que ele recomenda um aluno tutor para auxiliar o usuário na execução de uma tarefa.

Endereço http://localhost:8080/

Marcelo, porque você não conversa com Elisa Boff

Seja bem vindo(a),
Marcelo Seefeld

Logout

Apostila

- 1. Introdução
 - 1.1 Exemplo 1.1
 - 1.2 Exemplo 1.2
- 2. Expressões
- 3. Variáveis
 - 3.1 Exemplo 3.2

Algoritmo para preparar um bolo de fubá:

Ingredientes:

- ovos
- 250 gramas de margarina cremosa
- 2 xícaras (chá) de açúcar
- 1 xícara (chá) de farinha de trigo
- 1 xícara (chá) de fubá
- 200 gramas de coco ralado
- 1 copo de leite
- 1 colher (sopa) de fermento

Parar Agente Iniciar Agente

FIGURA 4 – Recomendação de Tutor

4. Considerações Finais

Este artigo apresentou um modelo de agente de interface sócio-afetivo capaz de interagir com usuários com o objetivo de motivar a formação de grupo e promover a aprendizagem colaborativa. Em Greer et al. (1998) os autores também propuseram um mecanismo de recomendação de tutores. No entanto, utilizaram uma abordagem diferente já que não enfatizaram a dimensão sócio-afetiva na construção do modelo do



usuário. Além de buscar identificar alunos que melhor possam cumprir a tarefa de tutores para determinada tarefa, o agente proposto também está apto a encontrar e recomendar conteúdos adequados para cada estudante.

Atualmente, evidências científicas começam a apontar que a presença de um personagem em uma interface pode aumentar a confiança do usuário (Rickenberg, 2000) e melhorar a comunicação entre homem e máquina através da introdução de estímulos sociais (De Angeli, 2001). No entanto, para que tais efeitos de interação possam ocorrer é necessário que os personagens tenham papéis e comportamentos adaptados à comunicação com cada usuário Picard (1997). Em nosso sistema, o agente Kurrupako não apenas memoriza toda a interação do usuário com o ambiente para lhe fazer recomendações personalizadas de conteúdo, como também identifica outros alunos que melhor se adaptam à categoria de tutor para determinada tarefa. Ao considerar características sócio-afetivas do usuário para modelar e selecionar suas ações, o agente Kurrupako adquire propriedades julgadas fundamentais na produção de comportamento verossímil (Hayes-Roth 1999).

Os experimentos realizados até então mostram a capacidade do assistente em despertar interesse e capturar a atenção dos usuários, servindo como um forte componente motivador para os alunos. Um de nossos próximos desafios está relacionado à inferência de estados afetivos do usuário a partir de dados coletados durante a interação. No momento, estamos utilizando questionários e controles de interfaces gráficas para indicar tais estados. Além disso, estamos interessados na utilização de outros modelos de assistentes com características gráficas mais ricas na busca de uma melhor comunicação com o usuário e na obtenção de resultados mais expressivos no que diz respeito à percepção do usuário das capacidades e utilidade dos agentes.

5. Referências

- ANDRE, E.; RIST, T. and MULLER, J. Employing AI methods to control the behavior of animated interface agents. **Applied Artificial Intelligence**, Volume 13, Numbers 4-5, May 1999, 415-448.
- BALL, G.; LING, D.; KURLANDER, D.; MILLER, J.; PUGH, D.; SKELLY, T.; STANKOSKY, A.; THIEL, D.; DANTZICH, M. V. and WAX, T. **Lifelike computer characters: the persona project at microsoft**. pages 191-222, 1997. Disponível em: <http://research.microsoft.com/research/pubs/view.aspx?pubid=439>.
- CONATI, C. Probabilistic assessment of user's emotions in educational games. **Journal of Applied Artificial Intelligence**. 16(7-8):555-575. 2002.
- COSTA, A. C. da R. A Teoria Piagetiana das Trocas Sociais e sua Aplicação aos Ambientes de Ensino-aprendizagem. In: **Informática na educação: teoria & prática**, vol.6, num.1, 2003.
- CRAIG, S. D.; GHOLSON, B. and DRISCOLL, D. M. Animated Pedagogical Agents in Multimedia Educational Environments. In: **Journal of Educational Psychology**, 2002, Vol. 94, No. 2, 428-434.
- DE ANGELI, A.; LYNCH, P. and JOHNSON, G. Personifying the e-market: A framework for social agents. In: **Interact 2001 - Eighth IFIP TC.13 Conference on Human-Computer Interaction**, Tokyo, Japan, July 9-13, 2001.
- ELLIOTT, C.D.; RICKEL, J. and LESTER, J. Lifelike Pedagogical Agents and Affective Computing: An Exploratory Synthesis. In: **M. Wooldridge and M. Veloso**,

editors, **Artificial Intelligence Today**, number 1600 in Lecture Notes in Computer Science, pages 195--212. Springer-Verlag, 1999.

GREER, J. E.; MCCALLA, G. I.; COOKE, J. E.; COLLINS, J.; Kumar, V. S.; Bishop, A. and Vassileva, J. I. "The Intelligent Helpdesk: Supporting Peer-Help in a University Course", in **Proc. of the International Intelligent Tutoring Systems Conference (ITS'98)**, San Antonio, TX, 1998, pp. 494-505.

HAYES-ROTH, B.; JOHNSON, V.; GENT, R. V.; and WESCOURT, K. Staffing the web with interactive characters. **Commun. ACM**, 42(3):103-105. 1999.

LESTER, J. C., VOERMAN, J. L., TOWNS, S. G., and CALLAWAY, C. B. (1997). Cosmo: A life-like animated pedagogical agent with deictic believability. In **Proc. of the IJCAI97 Workshop on Animated Interface Agents: Making them Intelligent**. Nagoya, Japan, August 23-29.

MATURANA, H. e VARELA, F. **Árvore do conhecimento**. Ed. Psy, Campinas, 1995.

PICARD, R. **Affective Computing**. Cambridge: MIT Press, 1997. 262 p.

REATEGUI, E.; TORRES, R.; CAMPBELL, J. A. Using Item Descriptors in Recommender Systems. In: Nineteenth National Conference on Artificial Intelligence, 2004, San José, CA, Estados Unidos. **Proceedings of the Workshop on Semantic Web Personalization - AAI04**, 2004.

REATEGUI, E.; BOFF, E.; VICCARI, R. M. Proposta e Avaliação Preliminar de um Assistente Virtual para Recomendação de Conteúdos. In: **Simpósio Brasileiro de Informática na Educação, 2005**, Juiz de Fora. Proceedings XVI Simpósio Brasileiro de Informática na Educação, 2005.

RICKENBERG, R. and REEVES, B. The Effects of Animated Characters on Anxiety, Task Performance, and Evaluations of User Interfaces. In **Proceedings of CHI 2000 - Human Factors in Computing Systems**. The Hague, Amsterdam, The Netherlands. 1-6 APRIL 2000.

SHAW, E., JOHNSON, W. L., and GANESHAN, R. Pedagogical agents on the web. In: AGENTS '99: **Proceedings of the third annual conference on Autonomous Agents**, pages 283-290, New York, NY, USA. ACM Press.