

Modelagem do Aprendizado de Grupos de Alunos em Ambientes Colaborativos Utilizando *Data Mining*

Thereza Patrícia Pereira Padilha^{1,2}, Leandro Maciel Almeida¹, João Bosco da Mota Alves²

¹Centro Universitário Luterano de Palmas - CEULP
Caixa Postal 160 – CEP: 77.054-970 – Palmas – TO – Brasil
{thereza, leandro}@ulbra-to.br

²Pós-graduação em Ciência da Computação – INE/UFSC
CEP: 88.040-900 – Florianópolis – SC - Brasil
{thereza,jbosco}@inf.ufsc.br

Resumo. Este artigo descreve um estudo de modelagem do aprendizado de grupos de alunos em ambientes colaborativos utilizando técnicas de *Data Mining*. Um agente inteligente é proposto para extrair padrões a partir de uma base de dados que contém informações sobre as interações dos alunos entre si e com o ambiente colaborativo. A partir dos padrões extraídos, o professor pode ter uma noção mais concreta do que o grupo realmente está aprendendo, bem como possuir uma visão geral do processo ensino-aprendizado.

Palavras-chave: Agente Inteligente, *Data Mining*, Ambientes Colaborativos.

Workshop: II Workshop de Tecnología Informática Aplicada en Educación .

1. Introdução

A Informática na Educação tem evoluído bastante nessa última década, principalmente pelo desenvolvimento de ambientes/sistemas cada vez mais interativos. Além de possibilitar uma maior interação entre os alunos, o uso de tecnologias que auxiliem o processo de ensino-aprendizado como um todo tem se apresentado como uma boa alternativa como, por exemplo, a tecnologia de agentes inteligentes.

O termo agente tem sido bastante utilizado em diversas áreas da Ciência da Computação, principalmente na Inteligência Artificial (IA). Diante disso, existe uma considerável discussão para definir esse termo precisamente. Na literatura computacional, o termo agente é utilizado para determinar diversos tipos de programas que não precisam, necessariamente, apresentar um comportamento “inteligente” [Costa 1999]. Por outro lado, na literatura de IA, os agentes requerem esse tipo de comportamento. Apesar do comportamento inteligente ser mais complexo de implementar, oferece vantagens por ser capaz de tomar as suas próprias iniciativas com intuito de alcançar os seus objetivos.

Um dos maiores problemas identificados na Educação a Distância é a averiguação das atividades realizadas pelos alunos, referente a avaliação do que foi realmente aprendido por eles. Até o momento, os ambientes colaborativos existentes definem a modelagem do aprendizado do aluno analisando, por exemplo, a quantidade de acessos ao ambiente. Entretanto, a quantidade de acessos ao ambiente não reflete num possível aprendizado, de forma que um aluno pode acessar várias vezes o ambiente sem contribuir com as atividades.

Para acompanhar a evolução do aprendizado de grupos de alunos nos ambientes colaborativos é necessário armazenar um histórico das interações dos alunos entre si e dos alunos com o ambiente, analisando sua participação efetiva na execução das atividades. Sendo assim, um mecanismo automatizado para desempenhar essas atividades é de suma importância para auxiliar os professores no processo de ensino-aprendizado.

Diante dessa problemática, este artigo apresenta os primeiros resultados de um modelo de agente inteligente que fornece a modelagem do aprendizado de grupos de alunos em ambientes colaborativos a partir de técnicas de *Data Mining* (DM). Dessa forma, a seção 2 apresenta uma visão geral dos aspectos relacionados ao desenvolvimento de ambientes colaborativos. Na seção 3 são apresentados alguns conceitos relacionados aos agentes inteligentes. Na seção 4 é descrito o funcionamento da indução de regras, uma das técnicas de DM do paradigma simbólico. Na seção 5 é descrito o modelo do agente proposto e apresentado seu desempenho usando uma base de dados que contém informações das interações de grupos de alunos num ambiente colaborativo. Por fim, na seção 6 são apresentadas algumas considerações finais.

2. Ambientes Colaborativos

Os ambientes colaborativos, também conhecidos como *groupware*, são mais complexos de implementar, pois necessitam suportar vários aspectos para realizar uma sincronização das tarefas comuns dos usuários, que podem estar geograficamente separados. Dentre esses aspectos, destacam-se: visualização, percepção (*awareness*), arquitetura, definição de papéis, controle de concorrência e metodologia de aprendizado [Borghoff 2000].

A maioria dos ambientes segue as formas de visualização do estilo *What You See is What I See* (WYSWIS), sendo subdividido em *strict* e *relaxed*. O estilo WYSWIS *strict* é considerado mais inconveniente para um grande número de usuários, porque se um membro mover a tela do ambiente para o lado direito, por exemplo, todos os outros usuários terão a tela posicionada neste mesmo

lugar. O estilo WYSWIS *relaxed* não necessita que o estado da visualização seja exatamente o mesmo para todos os usuários. Assim, por exemplo, os usuários podem posicionar a tela do ambiente da sua forma preferida ou mover objetos sem distrair e/ou contrariar os outros usuários.

A percepção de usuários nos espaços compartilhados serve para socializar virtualmente o ambiente. Este aspecto, em geral, é implementado nos ambientes colaborativos apresentando uma lista dos usuários conectados.

A arquitetura dos ambientes colaborativos pode ser classificada como centralizada ou distribuída. A arquitetura centralizada é caracterizada pela execução da aplicação numa única máquina e que é acessada por várias máquinas (usuários), seguindo o modelo cliente-servidor. Embora esta arquitetura tenha um ponto de falha, isto é, se o servidor sair do ar todos os usuários ficam impossibilitados de trabalhar, a implementação é mais fácil se comparada à distribuída. Por outro lado, a arquitetura distribuída destaca-se pelo armazenamento de uma cópia da aplicação nas diversas máquinas. Apesar de cada usuário possuir a sua cópia da aplicação, é possível disponibilizar um aprendizado de modo síncrono, pois somente as alterações são trafegadas pela rede.

A definição de papéis em ambientes colaborativos auxilia a estruturar as interações entre os usuários do grupo, definir tarefas e gerenciar o acesso aos documentos compartilhados. Os papéis são, frequentemente, baseados nas habilidades, competências e conhecimentos do membro, podendo ser categorizados formalmente ou informalmente. Para grupos compostos por mais de seis usuários, a existência de um coordenador/moderador é importante.

O controle de concorrência visa fornecer aos usuários um espaço compartilhado consistente. Neste caso, tanto utilizando a arquitetura centralizada quanto distribuída, o servidor alterna a execução das tarefas dos usuários tão rapidamente que simula uma realização simultaneamente de tarefas, ou seja, realiza um pseudo-paralelismo. O propósito é que os usuários recebam as respostas logo após as suas solicitações em vez de esperar um longo tempo por resultados. O controle de concorrência pode ser dispensável quando a probabilidade de conflitos entre os usuários for baixa ou existir um moderador para coordenar as ações dos usuários.

Existem várias metodologias de aprendizado que estão implementadas em ambientes colaborativos. Em sua maioria, são metodologias que apresentam tarefas ou atividades para serem realizadas por um grupo de usuários/alunos. Dentre essas metodologias, destacam-se: resolução de problemas, estudo de casos e abordagem dos sete passos.

Enfim, há diversos ambientes colaborativos já implementados disponíveis na Web, tais como: *Learning from Case Studies* (LeCs) [Rosatelli et al. 2000], *Collaborative Learning Environment* (CLE) [TSC 2002] e *TeamWave Workplace* [Roseman 2003]. Muitos desses ambientes são, de certa forma, recentes na área e disponibilizam diversas ferramentas de comunicação entre os alunos.

3. Técnicas de *Data Mining*

Data Mining é um dos passos do processo de extração de dados, que tem como objetivo explorar e extrair padrões a partir de uma base de dados [Fayyad et al. 1996]. Para a extração desses padrões podem ser utilizadas várias técnicas, tais como: redes neurais, algoritmos de aprendizado simbólico (regras de produção, árvores de decisão, por exemplo), raciocínio probabilístico, entre outras.

Dentre as técnicas citadas, destaca-se o aprendizado simbólico que mede e calcula a quantidade de informação (*Gain Ratio*), baseando-se na Teoria Geral da Informação [Quinlan

1996]. Essa teoria possui como um de seus principais pontos a redução da entropia (desorganização). A entropia é uma grandeza que mede a desordem, tanto de objetos físicos quanto de informações. Quanto maior o grau da entropia maior é a desordem, ou seja, quanto menor o grau da entropia melhor a organização. Sendo assim, o melhor grau de entropia é que o que possui valor mais baixo [Arariboia 1989] [Monard et al. 1997].

Para calcular a entropia de um determinado atributo (coluna) da base de dados é necessário, inicialmente, calcular a entropia dos valores possíveis desse atributo a partir da seguinte fórmula:

$$E(A = v_j) = - \sum_{i=1}^n p(i) \log_2 p(i)$$

na qual, $A = v_j$ significa que o atributo A tem o valor v_j , n é o número de classes diferentes c_1, c_2, \dots, c_n , e $p(i)$ é a probabilidade de um registro pertencer à classe c_n . Após realizar o cálculo com todos os valores possíveis de um atributo, deve-se relacionar todos esses valores para encontrar o ganho de informação de um atributo a partir da seguinte fórmula:

$$Gain(D, T) = Info(D) - \sum_{i=1}^k \frac{|D_i|}{|D|} \times Info(D_i)$$

na qual, $Info(D)$ representa a entropia da base com um todo, k denota a quantidade de subconjuntos do atributo T , $Info(D_i)$ denota a entropia de cada subconjunto $D_1, D_2, D_3, \dots, D_N$, D_i denota a quantidade de exemplos de cada subconjunto e o D a quantidade de exemplos contidos na base.

A partir desses cálculos, pode-se identificar o atributo mais relevante para a construção das regras de produção. As regras geradas podem utilizar todos os atributos disponíveis na base ou somente alguns. O critério mais importante é construir regras de produção consistentes, isto é, regras que retratem um padrão de um subconjunto de registros.

4. Agentes Inteligentes

Segundo Russell e Norvig, um agente é uma entidade que pode perceber o seu ambiente através de sensores e agir nesse ambiente através de atuadores (*effectors*) [Russell and Norvig 1995]. Devido à inexistência de um conceito universal de agentes, Wooldridge e Jennings identificaram dois enfoques, de acordo com alguns atributos que estão associados a eles [Wooldridge 1995]:

- **noção fraca:** é aquela utilizada para denotar qualquer *hardware* ou sistema computacional que possua autonomia (operar sem intervenção direta de humanos ou de outros agentes), habilidade social (interagir com outros agentes através de algum tipo de linguagem de comunicação de agentes), reatividade (perceber seu ambiente e atuar de maneira oportuna às mudanças que ocorrem) e pró-atividade (ter a capacidade de tomar iniciativas para alcançar as suas metas, não atuando apenas sob percepção);
- **noção forte:** é aquela em que o termo agente tem um significado maior do que adotado na noção fraca, pois os pesquisadores desse tipo de agente tentam aplicar conceitos dos seres humanos como, por exemplo: crença, desejo e intenção.

Outros atributos considerados no contexto de agentes são: mobilidade (habilidade de se movimentar através de uma rede de computadores), cooperação (capacidade de trabalhar com outros agentes para realizar tarefas mais complexas) e aprendizagem (capacidade de aprender a partir de experiências passadas, modificando o comportamento diante das novas situações).

Como pôde ser observado, existe uma grande quantidade de atributos que os agentes podem possuir, porém implementar agentes com todos esses atributos não é uma tarefa trivial. Portanto, é preciso analisar o agente para verificar quais os atributos são realmente necessários, de acordo com o sistema que se pretende construir [Costa 1999]. Essencialmente, na literatura, existe um consenso entre os pesquisadores que a autonomia é o atributo central para a identificação de agentes [Weiss 1999].

5. Agente para Modelagem do Aprendizado do Grupo

O intuito do agente proposto é apresentar ao professor uma modelagem do desempenho do aprendizado dos grupos de alunos, num ambiente colaborativo para a resolução de problemas. Neste trabalho, o ambiente colaborativo escolhido para realizar os experimentos foi o RESOLVE, que está sendo desenvolvido no Laboratório de Inteligência Computacional do Centro Universitário Luterano de Palmas.

Esse ambiente possui chat, editor de texto colaborativo e votação como ferramentas de comunicação, além de disponibilizar um repositório para o compartilhamento de documentos. A metodologia de aprendizado adotada é a resolução de problemas, sendo constituída de cinco etapas [Padilha 2003]:

- **observação da realidade** - visão global do assunto a ser tratado, identificando um problema para o estudo;
- **pontos-chave** - identificação dos pontos-chave relacionados ao problema. É a construção de um modelo simplificado da estrutura do problema;
- **teorização** - explanação teórica sobre cada ponto-chave identificado através de leituras e pesquisas;
- **elaboração de hipóteses** - formulação de algumas hipóteses possíveis que solucionem o problema exposto;
- **aplicação à realidade** - eleição e aplicação da solução mais adequada.

Esse agente interage com uma base de dados que armazena as interações dos grupos de alunos entre si e com o ambiente, no processo da resolução de problemas. Como essa base de dados é atualizada constantemente, devido às interações dos alunos, a percepção desse agente é realizada durante a resolução de cada problema discutido.

A definição das variáveis a serem avaliadas teve como base os recursos disponíveis no ambiente colaborativo para a resolução de problemas utilizados e nos aspectos que evidenciam a aprendizagem. Esses aspectos podem estar relacionados à tomada de decisão, independência pessoal, falta de persistência na execução das atividades ou envolvimento com o grupo. Na Tabela 1 é apresentada a lista dessas variáveis, bem como uma breve descrição do seu significado.

Tabela 1: Variáveis da Base de Dados

Nome da Variável	Descrição da Variável
IdProblema	Identifica a participação na identificação do problema
IdGrupo	Identifica o grupo do aluno
Utilização do chat	Identifica a quantidade de acessos à ferramenta chat
Utilização do editor	Identifica a quantidade de acessos à ferramenta editor colaborativ
Utilização da votação	Identifica a quantidade de acessos à ferramenta votação
Disponibilização de material	Identifica a quantidade de materiais que os alunos disponibilizar repositório
Tomada de decisão	Identifica a iniciativa para o preenchimento dos formulários
Envolvimento com o grupo	Identifica a quantidade de interações efetuadas entre os alun grupo
Falta de persistência	Identifica a falta de persistência do grupo em resolver os problem

A partir dos valores das variáveis citadas, o agente predefine um status de aprendizado para a resolução do problema apresentada por cada grupo. Esse status possui três valores possíveis, que são: **Razoável**, **Bom** e **Ótimo**. Em seguida, esse status é adicionado à base de dados, sendo relacionado a cada problema resolvido. A análise da evolução do desempenho dos grupos, para a resolução dos problemas, é realizada considerando conjuntos de problemas, assim é possível ter um mapeamento superficial do desempenho dos grupos.

A fim de apresentar os primeiros resultados obtidos pelo agente proposto, foram realizados alguns experimentos utilizando uma base de dados contendo trinta registros. Cada registro é composto pelos valores das variáveis descritas anteriormente e pelo status, sendo separados por vírgula. Uma amostra desses registros pode ser visualizada na Figura 1. Esses valores foram originados a partir da experiência do uso dos ambientes colaborativos para a resolução de problemas.

```

1,4,6,1,3,1,3,12,1,Razoável
1,5,7,4,2,2,6,12,2,Bom
1,6,9,5,2,4,5,14,1,Ótimo
2,1,9,5,3,5,8,12,1,Bom
2,5,4,2,1,2,5,12,4,Bom
2,6,9,6,2,2,5,12,3,Ótimo
3,1,11,5,3,3,4,16,2,Ótimo
3,2,10,3,4,4,6,10,1,Bom
3,3,8,4,2,2,5,12,3,Bom

```

Figura 1: Amostra de Registros da Base de Dados.

O agente utiliza, como técnica de *Data Mining*, a indução de regras, sendo assim primeiramente é calculada a entropia de todas as variáveis da base, a fim de encontrar a melhor

variável para a construção das regras. Para exemplificar os cálculos de entropia será selecionada a variável tomada de decisão. Por se tratar de uma variável do tipo contínuo, é necessário primeiramente realizar uma discretização da variável com o objetivo de encontrar um melhor *threshold* (limiar) para particionar o conjunto [Quinlan 1993]. O *threshold* identificado para essa variável foi três, sendo assim existem subconjuntos com valores menores ou iguais a três e maiores do que três.

Após encontrar o *threshold*, os registros de cada subconjunto são agrupados em função do atributo classe (evolução do aprendizado). Os registros que estão contidos no subconjunto menores ou iguais a três pertencem a uma mesma classe (Razoável), portanto eles estão ordenados, possuindo entropia igual zero, $E(\text{tomada_de_decisão} \leq 3) = 0$. Por outro lado, os registros que estão contidos no subconjunto maiores do que três pertencem a três classes diferentes, ou seja, ainda há desorganização. Nesse caso, aplicando a primeira fórmula, apresentada na seção 3, tem-se:

$$E(\text{tomada_de_decisão} > 3) = -9/25 * \log_2(9/25) + -15/25 * \log_2(15/25) + -1/25 * \log_2(1/25) \\ \cong 0,352$$

no qual 9, 15 e 1 denotam o número de registros classificados em função dos seguintes valores do atributo classe Ótimo, Bom e Razoável, respectivamente.

o cálculo entre atributos que possuem muitos valores, já que a maioria ou todos subconjuntos gerados irá possuir apenas uma classe, e com $E(X)=0$, o ganho da informação por particionar esse conjunto será máxima, sendo isto uma inverdade. Assim, para aplicação do *split info*, calcula-se a entropia do atributo a partir da quantidade de valores de cada subconjunto, seguindo o exemplo:

$$E(\text{tomada_de_decisão}) = -25/30 * \log_2(25/30) + -5/30 * \log_2(5/30)$$

A fase posterior consiste no cálculo da entropia da base com um todo, ou seja, seleciona o atributo classe analisando a quantidade de registros para cada valor possível, que são: Bom - 15, Ótimo - 9 e Razoável - 6. Aplicando-se a primeira fórmula, tem-se o seguinte resultado:

$$E(\text{classe}) = -15/30 * \log_2(15/30) + -9/30 * \log_2(9/30) + -6/30 * \log_2(6/30) \\ \cong 0,450$$

Após a obtenção de todos os resultados parciais da primeira fórmula (entropia), aplica-se a segunda fórmula, tendo os seguintes valores:

$$\text{Gain}(D, \text{tomada_de_decisão}) = 0,450 - (5/30 * 0 + 25/30 * 0,352) \\ \cong 0,156$$

Por fim, aplicando o *split Info* tem-se:

$$\text{Gain Ratio}(D, \text{tomada_de_decisão}) = 0,156 / 0,198 \\ \cong 0,788$$

Enfim, realizando esse mesmo processo com as outras variáveis da base, têm-se os seguintes valores de *Gain Ratio*: Grupo $\cong 0,108$; Utilização do chat $\cong 0,467$; Utilização do editor $\cong 0,252$;

Utilização da votação \cong 0,198; Disponibilização de material \cong 0,097; Envolvimento do grupo \cong 0,053; Falta de persistência \cong 0,341. A partir dos resultados dos cálculos do ganho de informação, pôde-se observar que a variável Tomada de decisão foi considerada o atributo mais relevante da base, pois tem o valor mais alto entre os demais.

As regras de produção extraídas pelo agente, utilizando a base de dados mencionada, podem ser ilustradas na Figura 2. O valor da classe (Razoável, Bom e Ótimo), que corresponde ao desempenho do grupo, em cada regra é apresentado após a palavra reservada **ENTAO**.

```
SE Tomada de decisão <= 3 ENTAO Razoável
SE Tomada de decisão > 3 E
  Utilização do Chat > 10 ENTAO Ótimo
SE Tomada de decisão > 3 E
  Utilização do Chat <= 10 ENTAO Bom
```

Figura 2: Regras de Produção Geradas pelo Agente.

Com a extração das regras, o agente disponibiliza os padrões encontrados, isto é, padrões que caracterizam os status Razoável, Bom e Ótimo. Para identificar o desempenho de cada grupo, o agente realiza as seguintes tarefas:

1. verifica os valores possíveis da variável IdGrupo, neste exemplo são disponibilizados somente seis grupos;
2. para cada valor possível da variável IdGrupo seleciona um conjunto de registros que descrevem as interações feitas na resolução do problema, através da variável IdProblema. Esse conjunto de registros é variável de acordo com a quantidade de problemas submetidos, isto é, para quinze problemas apresentados, por exemplo, o agente seleciona cinco conjuntos de três registros cada;
3. em seguida, submete os valores das variáveis dos registros selecionados aos padrões (regras de produção) extraídos pelo agente para encontrar o seu status.

Dessa forma, o professor terá uma noção abrangente do desempenho dos grupos e, assim, proporcionará uma atenção maior aos grupos que estão tendo algumas dificuldades. Os status definidos são armazenados em um arquivo texto, identificando cada grupo e os seus respectivos status em cada conjunto de problemas. Na Figura 3 podem ser visualizados os primeiros status de desempenho de cada grupo, predefinidos pelo agente. Além disso, é informado também um valor de aproveitamento total, em porcentagem, de acordo com os status gerados.

```
Grupo: Status do Desempenho: Aproveitamento
Grupo1: BOM -> BOM -> RAZOÁVEL: 70%
Grupo2: RAZOÁVEL -> BOM -> ÓTIMO: 72%
Grupo3: BOM -> BOM -> ÓTIMO: 85%
Grupo4: RAZOÁVEL -> BOM -> RAZOÁVEL: 60%
Grupo5: BOM -> ÓTIMO -> ÓTIMO: 95%
Grupo6: ÓTIMO -> BOM -> BOM: 80%
```

Figura 3: Status do Desempenho dos Grupos.

Neste exemplo, a base de dados contém trinta registros informando as interações de seis grupos, na resolução de cinco problemas. Assim, foram criados três conjuntos de registros, formados por dois, dois e um registro, respectivamente. No caso do grupo 2, pode-se inferir que os alunos desse grupo estão se adaptando a metodologia e, provavelmente o aprendizado está em ascensão. Já o grupo 5, por exemplo, possui um aproveitamento de 95% na resolução dos problemas, segundo o agente, demonstrando um bom desempenho em relação aos demais.

Para se ter uma noção mais abrangente do status do desempenho, gerado para cada grupo, foi construído um gráfico que apresenta, em colunas, o andamento do grupo em cada conjunto de problemas, no qual pode ser ilustrado na Figura 4.

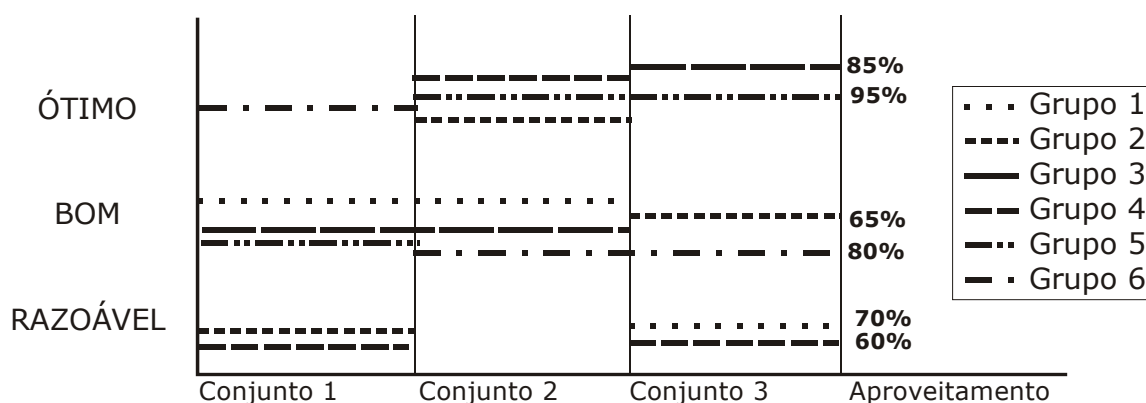


Figura 4: Visualização Gráfica do Status do Desempenho dos Grupos.

Para o uso do agente proposto num ambiente colaborativo, é preciso que se tenha a implementação de um agente responsável para capturar as interações dos alunos entre si e com o sistema e, em seguida, armazenar numa base de dados. Em geral, agentes que tem esse papel são denominados de agentes de interface e, na maioria das vezes, estão localizados na máquina do aluno.

6. Considerações Finais

A Educação a Distância ganhou um novo entusiasmo com o surgimento dos ambientes colaborativos. Entretanto, ainda há algumas carências no que diz respeito à atividade cognitiva, ferramentas para auxiliar o professor na avaliação do aluno ou no acompanhamento das atividades realizadas por grupos de alunos.

Com o agente inteligente proposto, espera-se auxiliar o professor na análise do aprendizado dos grupos de alunos num ambiente colaborativo para resolução de problemas. No entanto, esse agente pode ser aplicado a qualquer ambiente colaborativo que disponibilize uma base de dados com as interações dos usuários.

Através da modelagem realizada pelo agente, é possível identificar os conjuntos de problemas que os grupos tiveram mais dificuldades, mais facilidades e também grupos que se adaptaram a essa metodologia de aprendizagem com o passar do tempo. O uso de técnicas de DM foi considerado uma boa alternativa para identificar padrões de desempenho a partir das interações dos alunos armazenadas numa base.

Pôde-se verificar, nos experimentos realizados, que o uso desse agente possibilita ao professor uma visão geral do desempenho dos grupos na resolução de problemas. Sendo assim, a

sobrecarga de atividades a serem realizadas pelo professor pode ser amenizada principalmente no aspecto do acompanhamento do aprendizado do aluno.

7. Referências

- Borghoff, U. M. and Schlichter J. H. (2000) "Computer-Supported Cooperative Work: introduction to distributed applications", Springer.
- Fayyad, U., Shapiro, G. P. and Smyth, P. (1996) "From Data Mining to Knowledge Discovery in Databases". AAAIMIT Press, p.37-54.
- Costa, M. T. C. (1999) "Uma Arquitetura Baseada em Agentes para Suporte ao Ensino a Distância", Tese de Doutorado – PPGEF-UFSC.
- Grupo Arariboia.(1989) "Inteligência Artificial: um Curso Prático", Editora LTC, Rio de Janeiro.
- Monard, M. C., Batista, G. E., Kawamoto, S. and Pugliesi, J.B. (1997) "Uma Introdução ao Aprendizado Simbólico de Máquina por Exemplos", Nota Didática do ICMC-USP, n.º 20.
- Padilha, T. P. P. (2003) "Um Ambiente de Aprendizado Colaborativo para Resolução de Problemas", Monografia de Qualificação de Doutorado – UFSC.
- Quinlan, J. R. (1993) "C4.5: Programs for Machine Learning", San Mateo: Morgan Kaufmann.
- Quinlan, J. R. (1996) "Improved Use of Continuous Attributes in C4.5", Journal of Artificial Intelligence Research, vol.4, pp. 77-90.
- Rosatelli, M., Self, J. and Thiry M. (2000) "LeCS: A Collaborative Case Study System", Proceedings of the 5th International Conference on Intelligent Tutoring Systems, Montreal, Canada, pp. 242-251.
- Roseman, M. (2003) "TeamWave Workplace", Disponível em: <http://www.markroseman.com/teamwave/workplace.html>. Acesso em: 01/07/2003.
- Russell, S. J. and Norvig, P. (1995) "Artificial Intelligence: A Modern Approach", Englewood Cliffs, NJ, Prentice Hall.
- TSC (2002) "CLE: Collaborative Learning Environment", Disponível em: <http://www.aln.orgalnconf99/presentations/convertedfiles/63/sld001.htm>. Acesso em: 01/07/2003.
- Weiss, G. (1999) "Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence", MIT Press, Cambridge.
- Wooldridge, M. and Jennings, N. R. (1995) "Intelligent Agents: Theory and Practice", The Knowledge Engineering Review, vol.10, n.º 2, pp.115-152. Cambridge University Press.