

Sistema Tutor Inteligente baseado em Agentes na plataforma MOODLE para Apoio as Atividades Pedagógicas da Universidade Aberta do Piauí

Samuel B. J. Silva, Vinicius Ponte Machado, Francisco N. C. Araújo

Departamento de Computação – Universidade Federal do Piauí (UFPI)
Campus Universitário Ministro Petrônio Portella - Bairro Ininga - Teresina – PI – Brasil

{samuelcompufpi,netoaraujjo}@gmail.com, vinicius@ufpi.edu.br

***Abstract.** This paper presents a proposal to create an Intelligent Tutoring System for the MOODLE platform, with the goal of helping the educational activities of the Open University of Piauí (UAPI). The implementation done using Intelligent Agents, who will make a classification of students each machine learning, identifying their groups for the learning program to suit the student profile and find the best teaching methodology to be used for each.*

***Resumo.** Este artigo apresenta a proposta de criação de um Sistema Tutor Inteligente para a plataforma MOODLE, com o objetivo de ajudar nas atividades pedagógicas da Universidade Aberta do Piauí (UAPI). A implementação feita usando Agentes Inteligentes, que farão uma classificação dos estudantes cada aprendizado de máquina, identificando seus grupos para que o programa de aprendizado possa se adequar ao perfil do discente e encontrar a melhor metodologia pedagógica a ser utilizada para cada um.*

1. Introdução

O avanço no processo de ensino-aprendizagem evoluiu ao ponto de se ter, atualmente, três modalidades de ensino. São eles: presencial, semipresencial e a distância. A modalidade a distância se distingue do modo tradicional de ensino pelo fato de ser mediada por tecnologia. Geralmente, um curso à distância necessita de um Ambiente Virtual de Aprendizagem (AVA) que é um software que auxilia na montagem de turmas acessíveis pela Internet. Desenvolvido para ajudar os professores no gerenciamento de conteúdos para seus alunos e na administração do curso, permite acompanhar o progresso dos estudantes. O MOODLE (*Modular Object Oriented Dynamic Learning Environment*) é um software livre e seu desenvolvimento objetiva o gerenciamento do aprendizado, além de ser considerado por seus usuários uma ferramenta de fácil utilização devido à comunicação entre aluno e professor por ele fornecida. Entretanto, geralmente o professor não pode auxiliar o aluno em uma atividade em tempo hábil, pelo fato de ambos utilizarem utilizando a ferramenta simultaneamente. Isso impede, por muitas vezes, que o professor (tutor) possa auxiliar em uma tarefa ou até mesmo indicar uma atividade para o aluno em tempo hábil.

O objetivo deste artigo é apresentar um Sistema Tutor Inteligente (STI) com o intuito de aprimorar o ensino mediado por tecnologia, fazendo a intermediação entre o personagem que busca o aprendizado e uma ferramenta facilitadora do aprendizado, no caso o MOODLE/UAPI, em uso pela Universidade Aberta do Piauí (UAPI).

2. Referencial Teórico

2.1. Ambientes Virtuais de Aprendizagem (AVAs)

A plataforma de software na qual se baseiam as práticas de Educação a Distância (EaD) são chamados Ambientes Virtuais de Aprendizagem (AVAs). Por meio dessas ferramentas é que se disponibilizam os canais de interação, de acesso a conteúdo e o apoio prestado aos aprendizes em EaD [Schlemmer et. al. 2007].

Esses sistemas reproduzem a sala de aula para o meio virtual e buscam, além disso, usar a tecnologia para propiciar aos alunos novas possibilidades que promovam a aprendizagem. Suportam uma grande variedade de estilos de desenvolvimentos de capacidades, encorajando a colaboração e o aprendizado baseado em pesquisa, além de promover compartilhamento e o reuso de recursos.

2.2. Sistemas Tutores Inteligentes

Sistemas Tutores Inteligentes (STI) são programas de computador que auxiliam no processo de ensino-aprendizagem, utilizam técnicas de Inteligência Artificial (IA) e têm propósitos educacionais. STI pode simular o processo do pensamento humano para então auxiliar na resolução de alguns tipos de problemas [Rickel 1989].

Esses sistemas surgiram do aperfeiçoamento dos sistemas de Instrução Auxiliado por Computador (CAI) que se caracterizam por auxiliar o processo de ensino. Da instrução de técnicas e métodos de IA em CAI resultaram os Sistemas Tutores Inteligentes capazes de fornecer um ensino personalizado para os estudantes. É tarefa de um STI a apresentação correta do assunto em um tempo determinado pelo usuário (aluno), trabalhando dentro de um cronograma pré-estabelecido, gerando diagnóstico e evolução em tempo real [Rodrigues e Novais 2005].

Geralmente, a arquitetura básica dos STI é composta por quatro módulos: o módulo do domínio, que contém o conhecimento do conteúdo que será ensinado; o módulo modelo estudante (abordado neste trabalho) que contém as informações sobre o estudante usuário do STI; o módulo tutorial, que contém as estratégias de ensino; e o módulo da interface, responsável por administrar a interação do sistema com o estudante [Nwana 1990].

2.3. Agentes Inteligentes

Um agente inteligente é um sistema computacional que pode agir com independência e autonomia adotando a melhor ação possível diante de uma sequência de percepções atuando em um ambiente. Ele utiliza sensores, que são responsáveis pela percepção do ambiente e permitem ao agente acesso às informações referentes ao ambiente em que está atuando, e atuadores, que têm função de realizar ações nesse ambiente [Russell e Norvig 2004].

Um agente racional é aquele que obtém sucesso em suas atuações. Tal sucesso é determinado por um método de medida de desempenho. Medidas de desempenho são critérios para medir o sucesso do comportamento do agente no ambiente.

Vários trabalhos utilizam agentes no contexto de Sistema Tutor Inteligente [Damasceno e Cruz 2010] que apresenta uma proposta de arquitetura multiagente para o

STI PAT2Math (*Personal Affective Tutor To Math*). Este sistema tem como objetivo o ensino de álgebra elementar a alunos do ensino fundamental em um ambiente capaz de considerar as emoções do aluno. Em Giraffa (1999) é apresentado o MCOE que é um paradigma de agentes para modelar Sistemas Tutores Inteligentes com interface e funcionamento baseados em técnicas de programação de jogos.

2.4. Aprendizagem de Máquina

Aprendizado de Máquina é o ramo da Inteligência Artificial que estuda o desenvolvimento de sistemas, que são capazes de aprender com a experiência. Isto é, um sistema que realiza determinada tarefa e que possa obter informações que permitam execuções futuras desta tarefa com um melhor desempenho [Mitchell 1997]. Existem três principais tipos de técnicas de aprendizagem de máquina:

- Aprendizado Supervisionado: Implica necessariamente a existência de dados de entrada e a indicação de uma saída que possa ser aprendida para ocorrer o processo de aprendizagem [Braga et al 2007].
- Aprendizado Não-Supervisionado: envolve a aprendizagem de padrões na entrada, quando não são fornecidos valores de saídas específicas [Russell e Norvig 2004].
- Aprendizado por Reforço: se dá através de um agente, que toma decisões, em um ambiente e é recompensado ou punido por suas ações, tendo por objetivo aprender a melhor sequência de decisões a tomar para acumular o maior número de recompensas [Sutton e Barto 1998].

Neste projeto, o agente desenvolvido utilizou aprendizagem de máquina para analisar o perfil do aluno e assim poder recomendar exercícios e tarefas de acordo com seu nível de desempenho, a fim de melhorar o aprendizado. Será utilizada a técnica de aprendizagem não-supervisionada permitindo, assim, classificar os alunos em grupos (*clusters*), tornando possível a melhor recomendação de atividades pedagógicas para cada perfil.

O algoritmo utilizado para a classificação de perfil de usuários foi o *k-means*, que tem por objetivo criar partições de uma população n-dimensional em uma dada base de dados. O algoritmo *k-means* utiliza um parâmetro de entrada *k*, que determina a quantidade de *clusters* (coleção de objetos que são similares uns aos outros (de acordo com algum critério de similaridade pré-definido) e dissimilares a objetos pertencentes a outros clusters), sendo que tais *clusters* possuem n elementos (os *clusters* podem ter quantidade de elementos diferentes). Após a execução pretende-se detectar uma alta similaridade dos elementos de um grupo e baixa similaridade entre os *clusters* criados pelo algoritmo. Outro algoritmo utilizado foi o J48, um algoritmo de aprendizado de máquina preditivo que decide o valor de destino (variável dependente) de uma nova amostra com base em vários valores de atributos dos dados disponíveis formando uma árvore de decisão. Os nós internos de uma árvore de decisão denotam os diferentes atributos; os ramos entre os nós nos dizem os possíveis valores que esses atributos podem ter nas amostras observadas, enquanto que os nós terminais nos dizem o valor final (classificação) da variável dependente [Foruzan et al 2013].

2.5. Weka

A ferramenta Weka contém uma coleção de algoritmos de aprendizado de máquina e ferramentas de pré-processamento de dados, projetada para que possa experimentar rapidamente os métodos existentes em novos conjuntos de dados de forma flexível. Ainda fornece suporte extensivo para todo o processo de mineração de dados experimental, incluindo a preparação dos dados de entrada, avaliação de sistemas de aprendizagem estatística, visualizar os dados de entrada e o resultado da aprendizagem. Essas ferramentas diversas e abrangentes são acessadas através de uma interface comum, de modo que seus usuários podem comparar diferentes métodos e identificar aqueles que são mais adequadas para o problema em questão [Frank e Witten 2005].

O sistema é escrito em Java e distribuído sob os termos da GNU *General Public License*. Ele executa em diversas plataformas e foi testado em Linux, Windows e sistemas operacionais Macintosh. O Weka fornece uma interface uniforme para muitos algoritmos de aprendizagem diferentes, juntamente com os métodos de pré e pós-processamento e para avaliar o resultado de sistemas de aprendizagem em qualquer conjunto de dados fornecido. Ele também inclui métodos para os problemas de mineração de dados padrão: regressão, classificação, *clustering*, de mineração de regras de associação e seleção de atributos.

Os algoritmos utilizados neste trabalho estão disponíveis na ferramenta WEKA (*Waikato Environment for Knowledge*) [Frank e Witten 2005]. O WEKA foi desenvolvido na Unidade de Waikato na Nova Zelândia e por ser implementado na linguagem Java tem como principal característica a portabilidade, podendo ser executada em diversas plataformas. É um software livre e de código aberto, tais características facilitaram a sua utilização na camada PHP/JAVA descrito na sessão 3.1.

3. Sistema Tutor Inteligente para a plataforma MOODLE-UAPI

A maioria dos STI clássicos se caracterizam por utilizar uma única estratégia para ensino, possuindo assim pouca versatilidade em seu comportamento pedagógico, sem proporcionar dinamismo às características individuais de aprendizagem de cada aluno. Para preencher essa lacuna e fomentar a pesquisa na área, a metodologia de desenvolvimento do STI – MOODLE/UAPI seguirá o paradigma de agentes inteligentes. Os agentes modificam suas bases de conhecimento, percebem as intervenções do aprendiz e são dotados da capacidade de aprender e adaptar suas estratégias de ensino mediante a interação com o aluno auxiliando o papel do professor (tutor).

Em AVA, geralmente, o acompanhamento do aluno pelo tutor é feito de forma assíncrona, ou seja, o tempo entre a execução das tarefas dos alunos e a avaliação destas atividades e atribuições de outras pelos tutores é diferente. Isso dependerá do momento em que ambos estejam utilizando a ferramenta. Para amenizar a diferença do tempo do aluno e tutor, assim como auxiliar nas escolhas das atividades pedagógicas, um grupo de agentes, com atividades distintas, será desenvolvido compondo uma arquitetura de agentes do STI – MOODLE/UAPI. Atualmente, com a busca para ganhar tempo, economizar gastos e aumentar a flexibilidade para entrar no processo de aprendizagem, esse sistema vêm com o intuito de aprimorar o ensino mediado por tecnologia, fazendo a intermediação entre o personagem que busca um aprendizado e uma ferramenta

facilitadora do aprendizado, no caso o MOODLE/UAPI, sem necessariamente a intervenção do tutor (pessoa).

O modelo de STI proposto (Figura 1) engloba três tipos de agentes:

1. Agentes de Perfil: Será responsável por captar o perfil do aluno, identificando suas deficiências e necessidades;
2. Agente de desempenho: Proporciona condições de decisão de que tarefa ou ação a ser executada;
3. Agente Comunicador: Servirá de elo entre processo do STI e o tutor, colocando este a par das atividades exercidas pelos alunos e sugerindo intervenções pedagógicas.

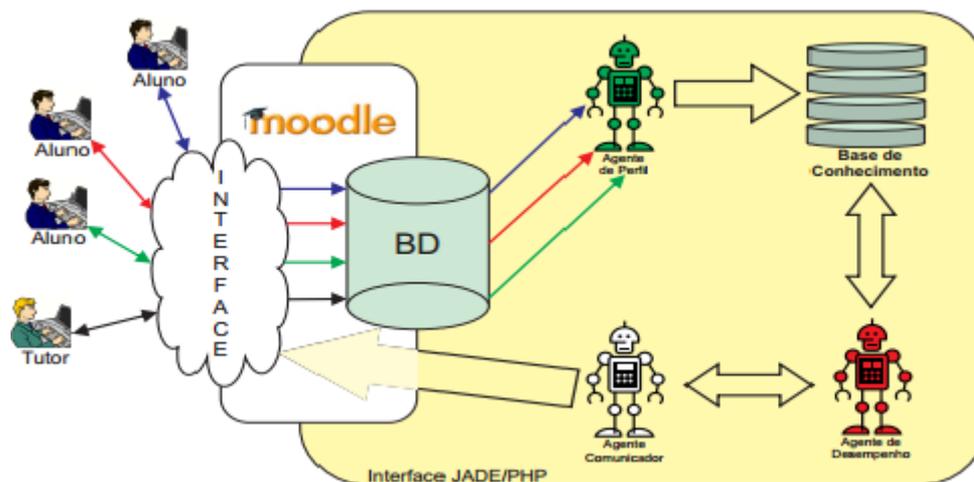


Figura 1. Arquitetura de Agentes para a plataforma MOODLE.

3.1 Interface JADE/PHP

Os agentes descritos nesta arquitetura são implementados em JADE (*Java Agent Development Framework*) [Bellifemine et. al. 2007] que é um *open-source* estruturado para tornar o desenvolvimento de aplicações multiagente mais rápido, em conformidade com as especificações FIPA (*Foundation for Intelligent Physical Agents*). O JADE pode ser considerado um *middleware* que implementa um *framework* de desenvolvimento e uma plataforma de agentes, acompanhando um conjunto de bibliotecas, para desenvolvimento de agentes na linguagem Java.

O MOODLE [Dougiamas e Taylor], acrônimo para (*Modular Object-Oriented Dynamic Learning Environment*), é uma plataforma de EaD, de utilização livre e códigos fonte aberto. Foi desenvolvido em PHP (um acrônimo recursivo para PHP: Hypertext Preprocessor), uma linguagem de *script* também de código aberto de uso geral, utilizada especialmente para o desenvolvimento de aplicações *web* dinâmicas, cujo código é executado no servidor.

Como se trata de duas tecnologias diferentes torna-se necessário a comunicação entre elas, fazendo com que os agentes possam interagir com as tarefas do MOODLE e possam acessar sua base de dados. Para solucionar esse problema foi desenvolvida uma camada de *software* que faz a comunicação dos agentes desenvolvidos em JADE e o MOODLE (PHP). Essa camada será responsável por “traduzir” as ações dos agentes

para plataforma MOODLE e permitir com que esses agentes possam realizar suas percepções sobre a interface e banco de dados do MOODLE.

3.2 Agentes

No projeto original foram utilizados três agentes com características distintas. O Agente de Perfil tem como função descobrir o perfil dos alunos; o Agente de Desempenho identifica as deficiências e necessidades de cada perfil proporcionando condições de decisão de qual ação pedagógica deve ser executada; e o Agente Comunicador servirá de elo entre processo STI e o tutor, lhe colocando a par das atividades exercidas pelos alunos e sugerindo intervenções pedagógicas.

4. Agente de Perfil (AP)

Neste artigo é apresentado o Agente de Perfil (AP) que será responsável por detectar o perfil dos alunos da plataforma MOODLE da UAPI. Através de algoritmos de *clustering* (técnica de aprendizagem de máquina para agrupamento (clustering) que exploram semelhanças entre padrões presentes em base de dados e os agrupam em categorias) e com base em atributos de desempenho (participação em fóruns, chats, exercícios, notas, etc), será criado grupos que classificam os alunos de acordo com sua participação nas atividades da plataforma MOODLE.

Nesta versão do STI, como mostra na figura 2, o agente de perfil cria uma base de dados baseada nas ferramentas de interação (*chat, upload, blog, course, forum, library, message, wiki, discussion*) de cada aluno encontrados na tabela *mdl_log* do MOODLE-UAPI que guarda os registros de atividades dos usuários no ambiente (figura 3). Logo após é feita a preparação dos dados (seleção dos atributos que serão utilizados na classificação dos grupos). Cada atributo é discretizado em quatro faixas de valores, utilizando a diferença entre a maior e a menor quantidade de participação de um atributo e dividindo por quatro para obter o valor base para calcular o valor de cada faixa.

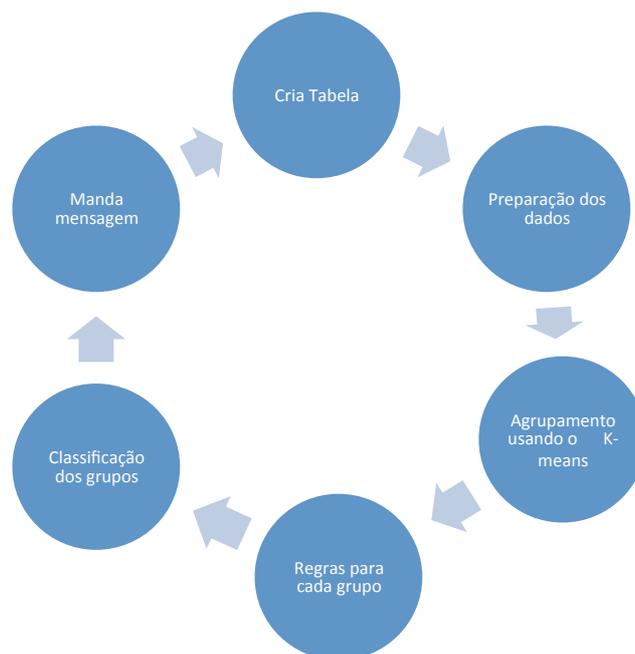


Figura 2. Fluxo de funcionamento do Agente de Perfil

atividades (10x2.210)									
usu...	chat	upload	blog	course	forum	library	message	wiki	discussion
4574	0	10	0	071	277	1000	5	0	2
4597	0	18	2	481	2473	1804	1	0	1
4600	0	17	23	2002	986	102	13	0	13
4601	0	14	3	632	2576	1750	8	0	1
4607	9	10	3	341	419	50	3	1	1
4608	4	19	1	372	1247	570	0	0	1
4609	37	21	0	673	977	66	4	7	1
4610	3	17	1	406	738	310	5	4	0
4611	4	16	3	334	470	54	19	3	2
4613	19	13	1	671	731	92	7	21	0
4614	10	1	0	110	112	65	5	3	0
4615	85	0	9	654	355	48	29	11	0
4616	111	17	13	1506	1966	1167	45	2	1
4618	7	3	3	372	159	49	1	1	0
4619	28	15	17	760	612	52	7	28	1
4620	25	1	1	520	1545	944	4	15	0
4623	18	1	1	251	111	49	13	1	0
4624	29	9	1	573	1315	13	2	1	0
4627	1	1	2	121	221	52	11	3	0
4628	12	9	0	561	378	51	1	4	0
4629	22	38	0	594	705	54	15	12	2
4630	62	30	4	2163	3019	51	8	22	1

Figura 3. Representação da tabela do bando de dados com a quantidade de participação de cada usuário nas ferramentas de interação do MOODLE

O algoritmo K-means (descrito na seção 2.4) é utilizado para classificar os alunos de cada turma da plataforma MOODLE-UAPI conforme sua participação em cada atributo. Para identificar quais foram as regras usadas para classificar os usuários em seus respectivos grupos foi utilizado o algoritmo J48 (descrito na seção 2.4). E com essas regras é feita a classificação dos grupos em ruim, regular, bom e ótimo. Finalizando, a fase de mandar a mensagem para os alunos e tutores fica por responsabilidade do agente de desempenho.

5. Funcionamento do AP

Para obtenção da classificação do perfil do aluno, foi utilizada a base de dados do ambiente virtual de aprendizagem (MOODLE), utilizado pela UAPI na oferta dos cursos da modalidade de Educação a Distância, em particular, a participação dos alunos do Curso de Graduação em Administração Pública com 2210 registros do ano de 2011. Como melhores atributos de classificação foram levados em consideração as contribuições efetivas de cada aluno, por disciplina, nas ferramentas de interação: *Chat*, *Upload*, *Blog*, *Course*, *Forum*, *Library*, *Message*, *Wiki*, *Discussion*. Com isso, o agente pode fornecer aos profissionais da educação envolvidos no processo de aprendizagem, informações acerca da participação dos alunos na disciplina de forma a fomentar ações pedagógicas diferenciadas.

Os alunos foram divididos em quatro grupos (ruim, regular, bom, ótimo) de acordo com a participação nas ferramentas de interação do MOODLE. As regras encontradas pelo J48 para classificação foram: Se a participação no *course* está na faixa 1 (0 até 5508) e participação no *wiki* está na faixa 1 (0 até 107) e participação no chat está na faixa 1 (0 até 404) então o usuário pertence ao grupo ruim; Se a participação no *course* está na faixa 2 (5509 até 11016) então o usuário pertence ao grupo regular; Se a participação no *course* está na faixa 1 (0 até 5508) e participação no *wiki* está na faixa 1 (0 até 107) e participação no chat está na faixa 2 (405 até 808) então o usuário pertence ao grupo bom; Se a participação no *course* está na faixa 1 (0 até 5508) e participação no *wiki* está na faixa 2 (108 até 214) então o usuário pertence ao grupo ótimo, como mostra a tabela 1.

Tabela 1. Características de cada grupo formado

G1: Ruim	G2: Regular	G3: Bom	G4: Ótimo
<i>course</i> = 1	<i>course</i> = 2	<i>course</i> = 1	<i>course</i> = 1
<i>wiki</i> = 1		<i>wiki</i> = 1	<i>wiki</i> = 1
<i>chat</i> = 1		<i>chat</i> = 2	

Uma vez executado o algoritmo, a classificação resultante servirá de subsídios para que o Agente de Desempenho (AD) possa, de acordo com a deficiência detectada pelas regras, através de seus atributos, incentivar participações no ambiente. Uma opção dessas ações poderia ser: mandar uma mensagem para todos os alunos pertencentes ao grupo que possuir pouca participação em diversos atributos para participar mais das ferramentas de interação *course* e *wiki* e principalmente do chat.

6. Conclusão

A proposta deste trabalho é incorporar um Sistema Tutor Inteligente baseado na tecnologia de agentes na plataforma MOODLE, em uso na UAPI para proporcionar melhorias no ensino dos cursos de educação à distância da Universidade Aberta do Piauí. Neste artigo apresentou-se o agente de perfil, responsável por criar uma classificação dos alunos com base em sua participação na plataforma MOODLE-UAPI. Através desta classificação os agentes de desempenho poderão sugerir atividades pedagógicas aos alunos, auxiliando o trabalho dos tutores.

Atualmente, somente o agente de perfil foi desenvolvido. Portanto, tem-se como trabalhos futuros o desenvolvimento dos demais agentes (desempenho e comunicador) e a construção da base de conhecimento. Com a implementação do STI por completo, o acompanhamento do aluno pelo tutor virtual (agente) estará sempre intervindo com o aluno, motivando sua participação. O tutor virtual encontrará uma melhor metodologia pedagógica, baseado em seu perfil, uma vez que será projetado para identificar o nível de ensino/aprendizagem de cada aluno. Portanto o STI proposto, baseado em agentes, é uma solução viável, ideal para ambientes virtuais de aprendizagem baseados na plataforma MOODLE.

7. Referências Bibliográficas

- Bellifemine, F. L. Caire, G. Greenwood, D. *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology)*. Wiley, April 2007. [Online]. Available: <http://www.worldcat.org/isbn/0470057475>
- Braga, A. P. Carvalho, A. P. L. F. Ludermir, T. B. “*Redes Neurais Artificiais: Teoria e Aplicações*”, 2nd ed., LTC, Ed. Rio de Janeiro, 2007.
- Damascento, P. J. F. Cruz, A. “Sistema tutor inteligente pat2math: Proposta de arquitetura multiagente,” *Workshop Escola de Sistemas de Agentes para Ambientes Colaborativos (WESAAC)*, 2010.
- Dougiamas, M. Taylor, P. C. “MOODLE: Using learning communities to create an open source course management system,” in *Proceedings of the EDMEDIA 2003 Conference, Honolulu, Hawaii, 2003*.
- Foruzan, K. Rouhollah, Mohammad, D. B. S. Mehdi, “Improving the methods of email classification based on words ontology”. *International Journal of Computer Science Issues*, Jul 2013
- Frank, E. and Witten, I. H, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd Ed., Elsevier, Ed. EUA, 2005.
- Giraffa, R. M. V. L. M. M. “Estratégias de ensino em sistemas tutores inteligentes modelados através da tecnologia de agentes,” *Revista Brasileira de Informática na Educação*, 1999.
- Mitchell, T. M. “Machine learning,” 1997.
- Nwana, H. S. *Intelligent tutoring systems: an overview*. Artificial Intelligence Review, s.1,4: 251-77, 1990.
- Rickel, J. “Intelligent computer aided instruction: a survey organized around system components,” IEEE. *Transactions on Systems, Man, and Cybernetics*, s.1., 19(1): 40-57, jan/feb 1989.
- Rodrigues, M. Novais, F. S. M. P. “Future challenges in intelligent tutoring systems: a framework,” *International Conference on Multimedia and Information and Communication Technologies in Education*, 2005.
- Russell, S. Norvig, P. *Inteligência Artificial*, Campus, Ed., 2004.
- Schlemmer, E.; Saccol, A. Z.; Garrido, S. Eliane, “Um modelo sistêmico de avaliação de softwares para educação a distância como apoio à gestão de ead”, *Revista de Gestão USP*, São Paulo, s.1., 14(1): 77-91, 2007.
- Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*, A. B. Book, Ed. Cambridge, 1998.