Um sistema multiagente de apoio à aprendizagem baseada em problema

Laysa Mabel de Oliveira Fontes¹ Francisco Milton Mendes Neto¹ Alexandre Ádames Alves Pontes²

Resumo: A aprendizagem baseada em problema (*Problem-Based Learning - PBL*) é uma teoria de aprendizagem que enfatiza a colaboração e o trabalho em grupo para resolução de um problema. No entanto, um problema que ocorre com frequência no processo de aplicação da PBL é a presença de conversações fora do contexto, que é uma situação na qual os estudantes perdem o foco das discussões do problema e passam a discutir outros assuntos. Outro aspecto importante é com relação ao processo de formação de grupos na PBL. Pode ser dificil para o facilitador alocar estudantes aos grupos a distância, uma vez que a ausência de um contato presencial dificulta o conhecimento de certas características importantes do perfil dos estudantes envolvidos no processo. Outro aspecto a ser considerado é a ausência de padronização dos conceitos relacionados à PBL, pois isso dificulta a compreensão comum e compartilhada sobre esse domínio. Dessa forma, este trabalho apresenta um sistema multiagente de apoio à PBL, com o objetivo de detectar e corrigir problemas inerentes à implantação desta teoria de aprendizagem. Para solucionar o problema de padronização dos conceitos relacionados à PBL, é proposta uma ontologia que preenche requisitos de abstração e significação dos conceitos relacionados ao domínio.

Palavras-chave: Aprendizagem baseada em problema. Sistema multiagente. Ontologia.

Abstract: The problem-based learning - PBL is a learning theory that emphasizes collaboration and teamwork to solve a problem. However, a problem that occurs frequently in the implementation of PBL is the out-of-context conversation, which is a situation in which the students lose focus and start talking about topics that are not related to the discussion. Another important aspect is related to the formation of groups in PBL. It might be difficult for the facilitator to assign students to groups at a distance, since the lack of presential contact makes it difficult to perceive important characteristics of the students' profiles involved in the process. Another aspect that must be assessed is the absence of standardization of PBL related concepts, for it makes their common and shared comprehension in the domain harder. Thus, this paper presents a multi-agent system to support PBL, with the objective of detecting and correcting problems inherent in the implementation of this learning theory. To solve the problem of standardization of PBL related concepts, we propose an ontology that meets the abstraction and significance requirements of concepts related to the domain.

Keywords: Problem-Based Learning. Multi-agent system. Ontology.

{alexandreadames@ufersa.edu.br}

http://dx.doi.org/10.5335/rbca.2011.1807

¹ Programa de Pós-Graduação em Ciência da Computação, Universidade do Estado do Rio Grande do Norte (UERN) / Universidade Federal Rural do Semi-Árido (UFERSA) - Mossoró (RN) - Brasil {laysa,miltonmendes@ufersa.edu.br}

² Superintendência de Tecnologia da Informação e Comunicação, UFERSA, BR 110 – Km 47 Bairro Pres. Costa e Silva - Mossoró (RN) - Brasil

1 Introdução

O Ensino a Distância (EaD) é uma modalidade de ensino e aprendizagem que tem crescido e apresentado bons resultados. Um dos grandes desafios atuais é o suporte informatizado a esta modalidade. Uma atividade importante existente no ensino presencial é a atividade em grupo. A interação entre os estudantes no desenvolvimento de alguma atividade pedagógica é fundamental para o processo de aprendizagem, pois cada estudante compartilha com os colegas seus conhecimentos, dúvidas e impressões sobre o que foi discutido na aula, enriquecendo, assim, o processo de aprendizagem e dando início a uma modalidade de aprendizagem, a Aprendizagem Colaborativa [1].

O suporte informatizado à aprendizagem colaborativa é chamado de Aprendizagem Colaborativa Apoiada por Computador, do inglês *Computer-Supported Collaborative Learning – CSCL* [2]. Os sistemas que implementam a aprendizagem colaborativa precisam incluir mecanismos de cooperação e comunicação para que o processo de aprendizagem possa ser efetuado com qualidade. Além disso, tais sistemas devem possuir mecanismos para criação de grupos e formas de acompanhamento pelos facilitadores (professores).

Na abordagem tradicional, o ensino é centrado no professor. O fundamento de tal postura é a de que o professor detém o conhecimento que deve ser transferido para o estudante. Nesse modo de ensinar, as atividades de aula se desenvolvem sob a única e exclusiva orientação do professor [3]. Na perspectiva da aprendizagem colaborativa, os estudantes trabalham juntos em pequenos grupos tendo uma única meta [1]. A aprendizagem é realizada de forma coletiva e o estudante é responsável por sua aprendizagem. A discussão de ideias entre os membros do grupo aumenta o interesse e promove o pensamento crítico.

A aprendizagem baseada em problema (*Problem-Based Learning - PBL*), segundo [4], é um método no qual os estudantes aprendem através da resolução de um problema, que em geral não possui uma solução trivial e uma única solução correta. A aprendizagem é centrada no estudante e o conhecimento é adquirido de forma autodirigida. Os estudantes trabalham em pequenos grupos colaborativos para identificar o que necessitam aprender para resolução do problema. O professor atua como facilitador do processo de aprendizagem, ao invés de apenas transmitir conhecimentos.

Agentes de software têm sido utilizados no âmbito educacional. Esta tecnologia tem se mostrado bastante promissora como auxílio em ambientes colaborativos de aprendizagem, dinamizando o processo. Eles podem ser usados, por exemplo, para auxiliar no cumprimento de uma dada teoria de aprendizagem em um ambiente colaborativo [5].

Outra tecnologia bastante utilizada em ambientes de suporte à aprendizagem são as ontologias. O uso de ontologias tem sido efetivo em diferentes aplicações ao solucionar algumas deficiências encontradas na representação do conhecimento de um domínio. Isso ocorre, principalmente, devido à sua característica de ser compartilhável e independente da aplicação, podendo, assim, ser utilizada por diversos sistemas.

Um problema que ocorre com frequência no processo de aplicação da PBL na EaD é a dispersão dos estudantes, causado, em grande parte, pela ausência física do professor para direcionar as discussões.

Outro aspecto importante nesse contexto é a formação de grupos. Na PBL, os membros de um grupo são responsáveis pela resolução de um problema e, para isso, precisam ter competências complementares em relação ao problema em questão. Além disso, pode ser dificil para o facilitador alocar estudantes aos grupos a distância, uma vez que a ausência de um contanto presencial dificulta o conhecimento de certas características importantes do perfil dos estudantes envolvidos no processo.

A ausência de padronização ou de uniformização dos conceitos relacionados à PBL também difículta a compreensão comum e compartilhada sobre este domínio.

Dessa forma, objetivando solucionar os problemas relacionados à dispersão dos estudantes e auxílio ao facilitador no processo de formação de grupos, é apresentado um sistema multiagente de apoio à PBL. Como forma de representar formalmente os conceitos relacionados ao domínio da PBL, é proposta uma ontologia que preenche os requisitos de abstração e significação dos conceitos relacionados ao domínio.

Este trabalho está organizado da seguinte forma: na Seção 2, são descritos os principais conceitos sobre a PBL; a Seção 3 apresenta uma explanação sobre sistemas multiagente; a Seção 4 descreve a problemática enfocada; na Seção 5, é apresentada a abordagem baseada em agentes de apoio à PBL; a Seção 6 descreve o

processo de desenvolvimento da ontologia proposta; na Seção 7, são apresentados os trabalhos relacionados; por fim, na Seção 8, são apresentadas as considerações finais e os trabalhos futuros.

2 Aprendizagem baseada em problema

Na PBL, o facilitador tem o papel de guiar os estudantes no processo de aprendizagem, identificando possíveis deficiências de conhecimento e habilidades necessárias à solução do problema proposto. Assim, nesta teoria de aprendizagem, ao invés de termos o professor simplesmente repassando os conhecimentos e depois testando-o através de avaliações, ele faz com que os estudantes apliquem o seu conhecimento em situações novas. Os estudantes se deparam com problemas muitas vezes complexos e que não possuem uma solução trivial e tentam descobrir, através da investigação e pesquisa, soluções úteis.

Para o sucesso da aplicação da PBL como estratégia pedagógica, os seguintes estágios devem ser cumpridos [4]: i) o facilitador propõe um problema complexo para o grupo de estudantes; ii) os estudantes tentam gerar fatos e identificar hipóteses sobre o problema através de um *brainstorming* inicial; iii) os estudantes formulam e analisam o problema, objetivando gerar ideias para sua solução; iv) os estudantes, auxiliados pelo facilitador, identificam deficiências de conhecimento para solução do problema; v) os estudantes procuram por novos conhecimentos relacionados ao domínio e tentam gerar fatos sobre este novo conhecimento; vi) ao final de cada problema, os estudantes refletem sobre os conhecimentos adquiridos. A Figura 1 ilustra o ciclo de desenvolvimento da PBL [4].

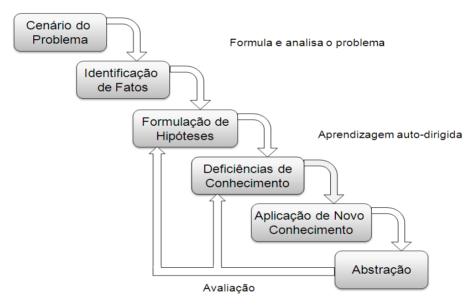


Figura 1. Ciclo da PBL

A PBL, quando aplicada, oferece alguns benefícios, dentre os quais se destacam [4]:

- desenvolve pensamento crítico e criatividade no estudante;
- aumenta sua capacidade de resolução de problemas;
- aumenta a motivação;
- ajuda os estudantes a aplicar os conhecimentos adquiridos em novas situações.

3 Sistemas multiagente

Segundo [6], agentes são entidades de software autônomas que percebem seu ambiente através de sensores e executam uma ação no ambiente através de atuadores, que processam informações e conhecimentos. Os Sistemas Multiagente (SMAs) podem modelar sistemas complexos, possibilitando que os agentes tenham objetivos comuns ou conflitantes. Estes agentes podem interagir entre si de duas formas: diretamente (via

comunicação e negociação) ou indiretamente (atuando sobre o ambiente). Os agentes podem cooperar para o benefício mútuo ou podem competir para servir seus próprios interesses [7]. Os sensores são as entradas de dados para o agente e os atuadores são os meios pelos quais o agente executa suas ações e interage com o ambiente. Uma sequência de percepções é tudo o que já foi captado pelo agente. Definimos o comportamento do agente por uma função de agente que mapeia qualquer sequência de percepções para determinada ação [6].

Existem diversos tipos de agentes: de software ou de hardware, estacionários ou móveis, persistentes ou temporários, reativos ou cognitivos. Uma das classificações mais clássicas sobre os agentes é quanto a serem reativos ou cognitivos.

Os agentes reativos são agentes simples que, ao perceberem alguma modificação no ambiente, reagem sem nenhum conhecimento de ações anteriores. Como esses agentes não possuem memória, são incapazes de planejar ações futuras. Agentes reativos simples selecionam uma ação baseados na percepção corrente do ambiente, ignorando percepções prévias. Os agentes cognitivos são agentes complexos, pois possuem uma representação explícita do ambiente e dos outros agentes. Este tipo de agente possui memória, o que o torna capaz de planejar ações futuras com base em situações ocorridas anteriormente [6].

Um meio-termo entre o agente reativo simples e o agente cognitivo é o agente reativo com estado interno, que para ter um desempenho mais racional salva um estado interno com aspectos de controle do mundo que não estão evidentes na percepção corrente. Este estado depende do histórico de percepções prévias do ambiente, sendo definido em um conjunto de possíveis estados internos correntes, $\Delta = \{\delta 1, ..., \delta n\}$. Essa estrutura de agente assume que: (1) o agente recebe informação, através de sensores, acerca dos estados do ambiente, definida em um conjunto de possíveis estados; (2) o agente possui um subsistema de percepção e um subsistema de tomada de decisão; e (3) o agente executa a ação selecionada no ambiente através de atuadores [5].

4 Problemática

A implantação de um método de ensino com base na PBL não é uma tarefa trivial. Em ambientes de aprendizagem colaborativa para EaD, a complexidade de implantação deste método é ainda maior, pois o facilitador nem sempre pode detectar possíveis problemas na colaboração, nem possui todas as informações necessárias para aplicar as técnicas de aprendizagem deste método [5].

A PBL enfatiza o trabalho em equipe como a chave para o sucesso do processo de aprendizagem. Em outras palavras, a colaboração é essencial [8]. Contudo, um problema frequente no processo de aprendizagem é a presença de conversações fora do contexto, nas quais os estudantes deixam de discutir aspectos relevantes do problema para discutir outros assuntos não relacionados ao contexto do problema. No ensino presencial, o professor pode detectar facilmente a presença desse problema e tentar corrigir essa situação para melhorar o processo de aprendizagem. No ensino a distância, por sua vez, isso não é uma tarefa trivial. Dependendo da quantidade de alunos e da variedade das ferramentas utilizadas para discussão, essa tarefa seria muito onerosa para o professor. O uso de agentes de software permite que o professor se concentre em outras atividades igualmente relevantes.

Outro aspecto importante é a formação de grupos. Na PBL, os membros de um grupo responsável pela resolução de um problema precisam ter competências complementares em relação ao problema que está sendo resolvido e pode ser difícil para o facilitador alocar estudantes aos grupos a distância, uma vez que a ausência de um contato presencial difículta o conhecimento de certas características importantes do perfil dos estudantes envolvidos no processo.

Outro problema importante a ser considerado é a ausência de padronização ou uniformização dos conceitos relacionados à PBL, pois isso dificulta a compreensão comum e compartilhada sobre este domínio. Não existe um mecanismo que torne possível a coleta em tempo hábil da informação realmente necessária para apoiar a tomada de decisão durante a resolução de problemas.

5 Abordagem baseada em agentes de apoio à PBL

A Figura 2 apresenta a arquitetura geral proposta neste trabalho.

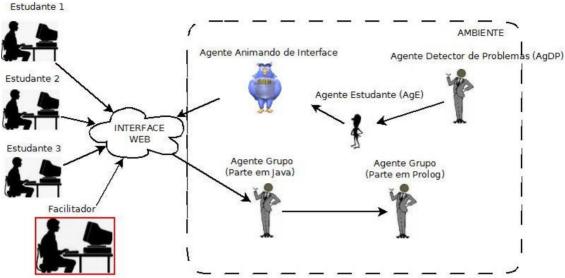


Figura 2. Arquitetura geral dos agentes

Nas subseções a seguir são apresentadas as abordagens baseadas em agentes para detecção de conversações fora do contexto e para formação de grupos na PBL.

5.1 Abordagem baseada em agentes para detecção de conversações fora do contexto na PBL

Agentes de software podem executar muitas tarefas na aprendizagem colaborativa apoiada por computador, tais como: monitorar a participação dos estudantes em discussões, facilitar a seleção de tópicos para discussão, avaliar a performance dos estudantes em relação ao uso de ferramentas de comunicação e cooperação disponíveis no ambiente, entre outras. O uso de agentes para auxiliar nessas tarefas está ficando cada vez mais importante, principalmente devido ao crescimento do número de estudantes que interagem em sistemas de apoio ao aprendizado, o que torna muito difícil para os facilitadores gerenciar essas atividades a distância. A abordagem para a detecção de conversações fora do contexto do problema está esboçada na Figura 3.

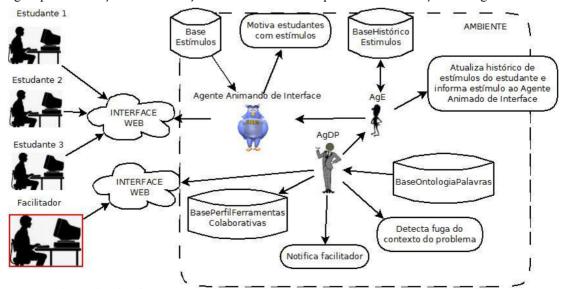


Figura 3. Abordagem para a detecção de conversações fora do contexto do problema

De acordo com a abordagem apresentada na Figura 3, três tipos de agentes são propostos: um Agente Detector de Problemas (AgDP), um Agente Estudante (AgE) e um Agente Animado de Interface.

A comunicação realizada entre os agentes na abordagem de detecção de conversação fora do contexto é apresentada no diagrama de sequência de mensagens na Figura 4.

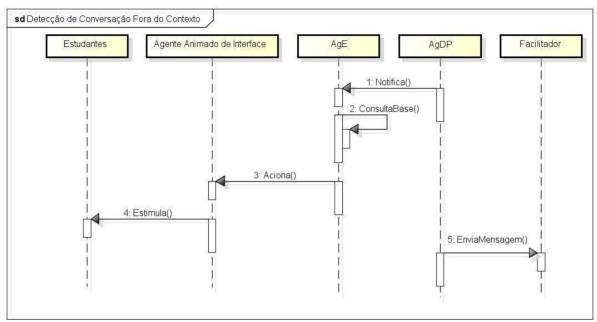


Figura 4. Diagrama de sequência de mensagens da abordagem de detecção de conversação fora do contexto

O AgDP é responsável por detectar as conversações fora de contexto baseado no uso das ferramentas de comunicação e cooperação disponíveis no ambiente e em uma ontologia do problema. Após detectar a fuga do contexto, o AgDP notifica o AgE, que consulta em sua base de históricos se este estudante foi estimulado antes.

Se o estudante não tiver sido previamente estimulado, o AgE irá acionar o agente animado de interface, que buscará o primeiro estímulo em sua base de estímulos. Um estímulo é uma mensagem textual mostrada pelo agente de interface ao usuário. Então, o agente animado de interface tentará motivar o estudante com este estímulo. Em seguida, o AgE gravará o tipo de estímulo usado na sua base de histórico e notificará o facilitador sobre a situação do estudante.

Contudo, se o AgE detectar em sua base de históricos que o estudante foi previamente estimulado, o AgE checará se já passou o prazo (previamente cadastrado pelo facilitador) que é dado para o estudante melhorar sua colaboração no ambiente de aprendizagem (ex. ao final de cada dia) discutindo sobre temas relacionados ao contexto do problema. Se o prazo não passou, o AgE não faz nada; se passou, o AgE notificará o agente animado de interface, que buscará o próximo estímulo em sua base de estímulos e notificará o facilitador sobre a situação do estudante.

Os agentes repetirão esse processo enquanto houver estímulos cadastrados na base de estímulos. O agente animado de interface é responsável por motivar os estudantes a não fugirem do contexto do problema durante as discussões e a usarem as ferramentas disponíveis no ambiente virtual de aprendizagem.

O AgDP é responsável por detectar estudantes que fugiram ao contexto durante as interações na PBL. O Algoritmo 1 apresenta os passos para alcançar essa meta.

Algoritmo 1 Ação: Detecção de fuga do contexto

Considerando:

- -> estudante: estudante resolvendo um mesmo problema na PBL
- -> ferramenta interação: algum mecanismo para colaboração síncrona ou assíncrona
- > mensagem: alguma mensagem enviada por um estudante através de alguma ferramenta de interação
- -> palavra: uma palavra em uma mensagem
- > OntologiaPalavrasRelacionadas: ontologia em que as palavras relacionadas ao contexto são
- -> PR: quantidade de palavras relacionadas ao contexto
- > OntologiaPalavrasNaoRelacionadas: ontologia em que as palavras não relacionadas ao contexto são instanciadas
- -> PNR: quantidade de palavras não relacionadas ao contexto
- $-> \Delta$: fator de balanceamento usado pelo facilitador
- -> NF: nível de fuga do contexto (guarda o resultado do algoritmo)

```
para todo estudante faça
1:
2:
      para todo mensagem € ferramenta interacao faça
3:
        para todo palavra € mensagem faça
4:
           se palavra € OntologiaPalavrasRelacionadas então
5:
                PR = PR + 1
6:
           senão
7:
                se palavra € OntologiaPalavrasNaoRelacionadas então
8:
                   PNR = PNR + 1
9:
                fim se
10:
           fim se
        fim para
11.
12:
      fim para
13:
      NF = PNR / (PR - \Delta)
14:
      return NR
15: fim para
```

Como pode ser visto no Algoritmo 1, o AgDP monitora constantemente as ferramentas usadas pelos estudantes para cooperação e comunicação durante a resolução do problema. Então, o AgDP compara as palavras usadas pelos estudantes nas suas interações com um conjunto de palavras previamente instanciadas em uma ontologia de palavras relacionadas e não relacionadas ao contexto do problema que está sendo resolvido.

Em seguida, o AgDP calcula a porcentagem de palavras usadas por estudante nas ferramentas disponíveis no ambiente fora do contexto do problema. Isso é útil para identificar o nível de fuga do contexto do estudante em relação aos assuntos relacionados ao problema em discussão. Este nível pode ser obtido pela expressão NF = PNR/(PR-Δ), onde NF = nível de fuga; PNR = quantidade de palavras não relacionadas; PR = quantidade de palavras relacionadas e Δ é um fator que o facilitador pode gerenciar para aumentar ou diminuir o impacto de palavras não relacionadas. Uma vez detectada uma conversação fora do contexto, caso o NF possua um valor superior ao definido pelo facilitador, o AgDP enviará uma mensagem automaticamente para o facilitador, via email, notificando o nome e o grupo dos estudantes dispersos. Dessa forma, o facilitador poderá tomar as providências cabíveis para manter os alunos focados no processo de resolução do problema em discussão.

5.2 Abordagem baseada em agentes para formação de grupos na PBL

A interação na PBL desempenha um papel muito importante no processo de aprendizagem. Nesse contexto, o processo de formação de grupos no ambiente de aprendizagem é de suma importância para o sucesso do processo como um todo. No ensino presencial, os estudantes estão próximos uns dos outros e, geralmente, o professor conhece cada um, assim como os estudantes se conhecem entre si. Na EaD, os estudantes estão geograficamente distribuídos; o facilitador normalmente não conhece os estudantes, tampouco eles se conhecem entre si. É papel do facilitador realizar a formação de grupos na PBL; contudo, na EaD ele não possui informação suficiente acerca dos estudantes para realizar esta tarefa sozinho. Na abordagem proposta, um agente

é utilizado para auxiliar o facilitador neste processo. O processo de formação de grupos proposto é ilustrado na Figura 5.

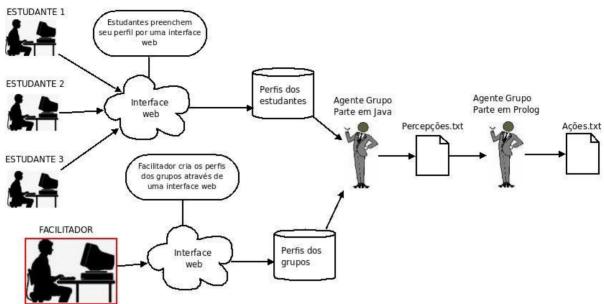


Figura 5. Processo de formação de grupos

O processo de formação de grupos é efetuado da seguinte forma: os estudantes, através de uma interface Web, preenchem seu perfil no início do processo. Esse processo alimenta uma base de perfis que será usada no processo de formação de grupos. O perfil dos estudantes é composto por habilidades, conhecimentos e deficiências, onde cada um possui um nível, que pode ser baixo, médio ou alto, podendo um estudante ter uma ou mais habilidades, deficiências e conhecimentos.

O facilitador preenche os perfis dos grupos desejados para cada problema a ser solucionado através de uma interface Web da mesma forma que o estudante. Um perfil de grupo é composto por habilidades, conhecimentos e deficiências, cada um possuindo um nível, que pode ser baixo, médio ou alto, bem como um valor fuzzy, que varia de 0.1 a 1.0 e está vinculado aos valores baixo, médio e alto. Após o facilitador construir os perfis dos grupos, haverá uma base de perfis de grupos que poderá ser consultada pelo Agente Formador de Grupos (AgFG) no processo de formação de grupos. É importante salientar que o perfil desejado construído pelo facilitador é o que melhor se adequa à resolução do problema; assim, um estudante que tenha um perfil aproximado ao desejado terá as competências necessárias à resolução do problema proposto.

O AgFG é o responsável pela formação automática dos grupos e foi construído usando duas linguagens: Java [9] e Prolog [10]. A parte do AgFG em Java é responsável pela geração de candidatos que estão aptos a participar de determinado grupo. Esse processo é feito analisando os perfis dos estudantes e os perfis dos grupos. Após essa análise, ele gera um arquivo, que será o arquivo de entrada para a parte do AgFG em Prolog.

A geração de candidatos é realizada da seguinte forma: (i) O AgFG analisa o perfil do grupo e verifica se um candidato possui uma habilidade presente no perfil do grupo; (ii) Se o candidato tiver pelo menos uma habilidade, é incluído numa lista de candidatos aptos a compor o grupo. Esse processo é feito de forma semelhante para conhecimentos e deficiências. Dessa forma, se o candidato possui, pelo menos, uma habilidade, conhecimento ou deficiência, é incluído na lista de candidatos aptos a participar do grupo; (iii) Em seguida, o AgFG gera uma entrada para a parte em Prolog. Essa entrada, chamada de percepção, é, na realidade, um arquivo que contém parâmetros que serão lidos pela parte do agente em Prolog, constituindo, assim, a interface entre as duas partes. Um exemplo da estrutura do arquivo pode ser visto na Tabela 1.

O primeiro parâmetro é a quantidade de candidatos; o segundo, um parâmetro de similaridade; o terceiro é a quantidade do universo de discurso; o quarto é a situação desejada (fontes e valores de importância) que representa o perfil do grupo com valores baixo (b), médio (m) e alto (a) e os valores fuzzy que variam de 0.1 a 1.0; por último, a situação corrente ou o perfil do(s) estudante(s) apto(s) a compor o grupo, ou seja, a quantidade

(1), o nome do estudante (Alexandre) e os valores baixo (b), médio (m) e alto (a) para as características do seu perfil.

Tabela 1. Exemplo da estrutura do arquivo

1º Parâmetro	1.
2º Parâmetro	2.
3º Parâmetro	11.
4º Parâmetro	[a,m,b,a,a]. [0.8,0.4,0.2,0.7,1.0].
5° Parâmetro	[1,'Alexandre',[m,m,b,m,b]].

A parte do AgFG em Prolog é responsável pela alocação dos estudantes aos grupos propriamente dita. No final desse processo é gerado um arquivo que contém os rankings para formação de grupos. O facilitador analisará este resultado, que é exibido através de uma página Web, e decidirá se acata ou não a sugestão do AgFG.

6 Processo de desenvolvimento da ontologia proposta

Na literatura são encontradas diversas definições para ontologia. Segundo [11], uma ontologia é uma especificação explícita de uma conceitualização.

Compreender o domínio de conhecimento que está sendo abordado é parte primordial na modelagem de uma ontologia. Definir conceitos, criar relações e definir uma hierarquia, ou seja, uma taxonomia entre os conceitos, requer, além do entendimento do assunto, a consciência do que se pretende a partir da iniciativa de criar uma ontologia.

Com o intuito de representar formalmente o domínio da PBL, foi feita uma análise deste domínio, na qual foi realizado um levantamento, com base na literatura, dos termos importantes e relevantes para o domínio. Feito isso, foi possível representar formalmente a ontologia de maneira específica, possibilitando o processamento e a abrangência do conhecimento não só por humanos, mas também por máquinas. Isso foi possível pelo uso de uma linguagem específica para a criação de ontologias e de uma ferramenta que permite sistematizar e integrar as especificações definidas à linguagem utilizada.

No desenvolvimento da ontologia proposta neste trabalho foi utilizada a linguagem OWL [12]. Esta linguagem de descrição de ontologia foi escolhida por apresentar características que tornam as ontologias mais robustas. A linguagem OWL apresenta também recursos adicionais não suportados por outras linguagens, como, por exemplo, sublinguagens incrementais, projetadas para serem usadas por diferentes comunidades de implementadores e usuários. A ferramenta selecionada para a modelagem da ontologia foi a Protégé [13].

As ontologias não apresentam sempre a mesma estrutura, mas existem características e componentes básicos comuns presentes em grande parte delas [14]. A ontologia proposta neste trabalho possui a seguinte estrutura:

- Classes e/ou subclasses: abrangem um conjunto de classes e uma hierarquia entre essas classes, ou seja, uma taxonomia:
- Propriedades: na ontologia proposta neste trabalho, foram definidos dois tipos de propriedades: *Object Property* e *Datatype Property*. Propriedades do tipo *Object Property* têm o papel de qualificar ou relacionar classes; já as Propriedades do tipo *Datatype Property* constituem campos que podem ser instanciados:
- Axiomas: modelam sentenças que são sempre verdadeiras. A criação de axiomas é feita através de definições formais.

6.1 Ontologia da aprendizagem baseada em problema

A seguir será apresentado o processo de desenvolvimento da ontologia sobre a PBL.

6.1.1 Principais classes identificadas

Fazendo uso da análise do domínio PBL, foram definidas 12 (doze) classes (Ciclo, Estrategias, Habilidades, Metas, Usuario, Grupo, Problema, Quadro_Branco, Fato, Ideia, Questao_de_Aprendizado e Plano_de_Acao), com o intuito de representar os conceitos gerais. A partir dos conceitos gerais foram definidas e agrupadas suas especificidades, ou seja, suas 20 subclasses. Por exemplo, para representar os termos mais específicos da classe Ciclo foram definidas 6 subclasses. Do mesmo modo, para as classes Estrategias, Habilidades, Metas e Usuario foram definidas 2, 4, 5 e 4 subclasses, respectivamente.

Como visto na Seção 2, o ciclo da PBL é composto de 6 fases: cenário do problema, identificação de fatos, formulação de hipóteses, deficiências de conhecimento, aplicação de novo conhecimento e abstração/reflexão [4]. A Figura 6 ilustra o desenvolvimento da classe Cenario_do_Problema na ferramenta Protégé, como forma de exemplificar o desenvolvimento de um conceito presente na ontologia proposta.



Figura 6. Representação da classe Cenario_do_Problema na ferramenta Protégé

Na Figura 6 temos a definição da classe Ciclo como sendo superclasse de Cenario_do_Problema. O trecho de código OWL referente a essa definição é exibido na Figura 7.

```
<owl:Class rdf:about="#Cenario_do_Problema">
    <rdfs:subClassOf rdf:resource="#Ciclo"/>
    </owl:Class>
```

Figura 7. Representação da classe Ciclo como superclasse de Cenario_do_Problema em OWL

O campo *Disjoint Classes*, ilustrado na Figura 6, é usado para desconectar um grupo de classes, ou seja, torná-las disjuntas. Isso garante que uma instância que tenha sido declarada como sendo membro de uma das classes do grupo não pode ser membro de nenhuma outra classe naquele mesmo grupo. Na Figura 6, temos todas as classes disjuntas à classe Cenario_do_Problema. A Figura 8 ilustra o código OWL referente à propriedade *Disjoint*.

```
<owl:class rdf:about="#Cenario_do_Problema">
  <owl:disjointWith>
        <owl:class rdf:about="#Aplicacao_de_Novo_Conhecimento"/>
        </owl:disjointWith>
        <owl:disjointWith rdf:resource="#Deficiencias_de_Conhecimento"/>
        <owl:disjointWith>
        <owl:class rdf:about="#Formulacao_de_Hipoteses"/>
        </owl:disjointWith>
        <owl:disjointWith>
        <owl:disjointWith>
        <owl:class rdf:about="#Identificacao_de_Fatos"/>
        </owl:disjointWith>
        <owl:disjointWith>
        <owl:disjointWith>
        <owl:disjointWith><owl:disjointWith rdf:resource="#Reflexao"/></owl:disjointWith rdf:resource="#Reflexao"/>
```

Figura 8. Representação das classes disjuntas à classe Cenario do Problema em OWL

Similarmente ao procedimento apresentado nesta subseção, foram desenvolvidas todas as outras classes que compõem o domínio em questão.

6.1.2 Especificação das propriedades das classes

As propriedades de cada classe da ontologia foram especificadas à medida que as classes foram sendo definidas ou reutilizadas. A ontologia resultante possui 32 propriedades do tipo *Object Property* e 15 propriedades do tipo *Datatype Property*.

Para representar uma propriedade do tipo *Object Property* ou *Datatype Property* é necessária a definição do domínio da propriedade (*domain*) e da classe à qual se aplica a propriedade (*range*). Uma propriedade deve ser declarada como *Object Property* quando tem o papel de qualificar ou relacionar classes. Podemos citar como exemplo do tipo *Object Property* a propriedade temEstrategia, que tem como domínio a classe Facilitador e, como *range*, a classe Estrategias. Esta propriedade foi criada para associar todas as estratégias que um dado facilitador possui, como, por exemplo, questionários e resumos. A Figura 9 ilustra a especificação da propriedade temEstrategia do tipo *Object Property* na ferramenta Protégé.



Figura 9. Representação da propriedade temEstrategia na ferramenta Protégé

O trecho de código OWL referente à propriedade temEstrategia é mostrado na Figura 10.

```
<owl:ObjectProperty rdf:ID="temEstrategia">
  <rdfs:range rdf:resource="#Estrategias"/>
  <rdfs:domain rdf:resource="#Facilitador"/>
  </owl:ObjectProperty>
```

Figura 10. Representação da propriedade temEstrategia em OWL

Propriedades do tipo *Datatype Property* constituem os atributos de uma classe que podem ser instanciados através, por exemplo, do preenchimento de campos em um formulário de entrada de dados. Neste tipo de propriedade é necessário definir o domínio ao qual ela pertence e a *range* que, diferentemente do tipo *Object Property*, não mais será uma classe, mas um tipo de dado (ex. *date, string, boolean, int, double*, entre outros). Podemos citar, como exemplo de propriedade do tipo *Datatype Property* a propriedade quantidadeDeMembrosDoGrupo, que tem como domínio a classe Grupo e, como *range*, o tipo *int*. A Figura 11 ilustra a especificação da propriedade quantidadeDeMembrosDoGrupo do tipo *Datatype Property* na ferramenta Protégé.

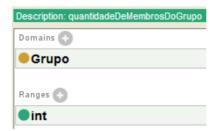


Figura 11. Representação da propriedade quantidadeDeMembrosDoGrupo na ferramenta Protégé

O trecho de código OWL referente à propriedade quantidadeDeMembrosDoGrupo é mostrado na Figura 12.

```
<owl:DatatypeProperty rdf:ID="quantidadeDeMembrosDoGrupo">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
    <rdfs:domain rdf:resource="#Grupo"/>
    </owl:DatatypeProperty>
```

Figura 12. Representação da propriedade quantidadeDeMembrosDoGrupo em OWL

O desenvolvimento de todas as outras propriedades que compõem a ontologia sobre a PBL seguiu o mesmo procedimento apresentado anteriormente.

7 Trabalhos relacionados

Os SMAs têm sido amplamente utilizados no âmbito educacional. Esta tecnologia tem se mostrado bastante promissora como auxílio em ambientes colaborativos de aprendizagem, tornando estes ambientes mais proativos e autônomos. SMAs podem ser usados, por exemplo, para auxiliar na implementação de uma dada teoria de aprendizagem em um ambiente colaborativo. Ontologias também têm sido bastante utilizadas em ambientes de suporte à aprendizagem, principalmente por sua capacidade de solucionar deficiências encontradas na representação do conhecimento de um domínio.

Em [15], é descrita uma arquitetura, baseada em um estudante simulado, que foi projetada para detectar e evitar três situações que podem dificultar o processo de aprendizagem em ambientes virtuais de aprendizagem: conversações que fogem ao conteúdo ministrado, estudantes com comportamento passivo e problemas relacionados ao aprendizado do estudante.

Em [16] é descrito um modelo para ambientes virtuais de aprendizagem que emprega agentes inteligentes para implementar a teoria sociocultural de Vygotsky, enfocando o aspecto social de interação. O modelo proposto possui diversos agentes, dentre os quais se pode destacar: (i) o agente social, que tem como principal objetivo a construção de modelos para os grupos de estudantes, além de identificar grupos de estudantes que podem cooperar em boas condições; (ii) o agente tutor, que avalia os objetivos educacionais do estudante e recomenda algum tipo de atividade; (iii) os agentes de assistência ao estudante, que monitoram suas atividades e se comunicam com os outros agentes.

Em [17] é proposta uma abordagem baseada em algoritmos genéticos para a formação dos grupos na qual são aplicados fatores de aceitação de um grupo pelo professor, levando em consideração o perfil dos alunos e a coesão do grupo, utilizando técnicas sociométricas. Esta abordagem é usada no ambiente NetClass de ensino-aprendizagem cooperativa.

Em [18] é apresentada uma aplicação de técnicas de inteligência artificial, mais especificamente, Sistemas Multiagente, para a formação de grupos colaborativos em um Ambiente Multiagente Interativo de Aprendizagem na web. A definição e implementação de uma arquitetura de agentes, modelados com algoritmos genéticos, são apresentadas, bem como sua integração com o ambiente TelEduc.

Em [19] é apresentada uma ferramenta, Smart Chat Group, que usa uma sociedade de agentes inteligentes para fazer acompanhamento, sugestão e formação automática de pequenos grupos de aprendizagem com base em informações de contexto dos aprendizes.

Em [20] é apresentada uma ontologia para buscar, descobrir e publicar materiais de aprendizado relevantes, como Objetos de Aprendizagem, para ajudar estudantes na abordagem da PBL.

Em [21] é proposta uma ontologia de domínio para a aprendizagem cooperativa, com o objetivo de fornecer uma conceituação explícita sobre esses elementos, ajudar outras pessoas a compreenderem melhor essa área de conhecimento e contribuir para a construção de ambientes cooperativos mais fundamentados.

Como diferencial do nosso trabalho podemos destacar: (i) a arquitetura proposta neste trabalho faz uso de um agente animado de interface, que apresenta características socioafetivas, ou seja, uma vez que o Agente Detector de Problemas identifica conversações fora do contexto do problema, o Agente Animado de Interface tenta resolver ou minimizar o problema motivando o aluno a participar das atividades e discussões. Para isso, ele utiliza mensagens textuais. (ii) o Agente Formador de Grupos, proposto neste trabalho, realiza a formação de grupos baseado tanto nos perfis dos estudantes como nos perfis dos grupos, sendo estes últimos criados pelo facilitador. (iii) Diferentemente dos outros trabalhos discutidos nessa seção, a abordagem proposta tem como base a PBL, que é uma teoria de aprendizagem comprovadamente eficaz [22] [23] [24]. Além disso, todos os

trabalhos abordados nesta seção diferem da proposta apresentada neste artigo, por usarem ontologias que não representam domínios de conhecimento referentes às teorias de aprendizagem, como a proposta deste trabalho, que é modelar uma ontologia de domínio para a PBL.

8 Considerações finais e trabalhos futuros

A PBL é uma teoria de aprendizagem que tem sido aplicada com sucesso em ambientes virtuais de aprendizagem. Esta teoria enfatiza o trabalho em equipe e a colaboração na resolução de problemas. Contudo, um problema que ocorre com frequência é a dispersão dos estudantes nas discussões que ocorrem em ambientes de aprendizagem colaborativa, que acabam influenciando a produtividade dos mesmos de forma significativa. Outro problema importante é com relação ao processo de formação de grupos na PBL. Pode ser difícil para o facilitador alocar estudantes aos grupos a distância, uma vez que a ausência de um contato presencial difículta o conhecimento de certas características importantes do perfil dos estudantes envolvidos no processo. Como não há contato presencial nesses ambientes, uma boa opção para reduzir o impacto desses problemas é o uso de agentes de software.

Este trabalho apresentou uma abordagem que faz uso de agentes de software para contornar o problema da perda de foco dos estudantes durante as interações com outros estudantes e para auxiliar a formação de grupos, provendo suporte ao facilitador para resolver esses problemas. Com a abordagem de agentes proposta, obtém-se uma diminuição na dispersão dos alunos, pois, ao detectar que a discussão nas ferramentas colaborativas está se distanciando do contexto em questão, o facilitador é notificado automaticamente sobre os estudantes dispersos, podendo, assim, tomar as providências cabíveis. A arquitetura também provê suporte ao facilitador para o processo de formação de grupos.

Para prover um maior esclarecimento sobre o domínio da PBL, não encontrado nos trabalhos pesquisados na literatura, foi apresentada uma ontologia que modela os relacionamentos entre os conceitos deste domínio. A ontologia proposta tem como diferencial sua abrangência, uma vez que foram contemplados: (i) termos relevantes ao domínio; (ii) as fases que caracterizam o ciclo da PBL; (iii) as estratégias que podem ser desempenhadas pelo facilitador na PBL; (iv) as habilidades que os estudantes devem adquirir no processo de aplicação da PBL, como, por exemplo, autoaprendizado, capacidade de resolução de problemas, colaboração e motivação; e (v) as metas relacionadas ao aprendizado dos estudantes na teoria de ensino-aprendizagem da PBL, dentre outros conceitos. A ontologia proposta também apresenta o detalhamento das principais ferramentas presentes na PBL, mostrando suas finalidades e por quem podem ser utilizadas.

Como contribuição deste trabalho podemos destacar o desenvolvimento de uma abordagem baseada em agentes de apoio à PBL, no que se diz respeito à detecção de conversações fora do contexto e auxílio ao facilitador no processo de formação de grupos. Dessa forma, a abordagem oferecerá um subsídio para que o facilitador possa acompanhar o processo de aplicação da PBL, possibilitando uma intervenção eficaz quando os problemas de aprendizagem, abordados neste trabalho, forem detectados. Por fim, a ontologia apresentada neste trabalho provê um esclarecimento sobre o domínio da PBL, servindo como uma ferramenta de consulta de conhecimento sobre este domínio que poderá ser utilizada em diferentes contextos.

Como trabalhos futuros, pretende-se melhorar a abordagem de detecção de conversação fora do contexto e o processo de formação de grupo apresentados neste trabalho, através de estudos na literatura sobre outras possíveis abordagens de soluções para os referidos problemas. Também se pretende abordar outras metas relacionadas ao auxílio no cumprimento da aplicação da PBL, conforme apresentado em [5]. Por fim, objetiva-se realizar um estudo de caso como forma de validar a eficácia da solução apresentada neste trabalho e uma análise quantitativa para obter dados estatísticos que comprovem a eficácia das soluções propostas.

Referências

- [1] COUTINHO, C. M. P.; BOTTENTUIT JUNIOR, J. B. Collaborative learning using Wiki: a pilot study with master students in educational technology In Portugal. In: PROCEEDINGS OF WORLD CONFERENCE ON EDUCATIONAL MULTIMEDIA, HYPERMEDIA, AND TELECOMMUNICATIONS. Vancouver, Canadá: [s.n.], p. 1786-1791, 2007.
- [2] DIMITRACOPOULOU, A. Designing collaborative learning systems: current trends & future research agenda. In: PROCEEDINGS OF THE CONFERENCE ON COMPUTER SUPPORT FOR COLLABORATIVE LEARNING: LEARNING 2005: THE NEXT 10 YEARS! [s.l.]: p. 115-124, 2005.
- [3] MARQUES, J. C. Aprendizagem Colaborativa: Atividades de Grupo como Núcleo e Uso do Computador como Contexto. Psicologia Argumento, v. 24, n. 44, p. 37-43, 2006.
- [4] HMELO-SILVER, C. E. Problem-Based Learning: What and How Do Students Learn? Educational Psychology Review, v. 16, n. 3, 2004.
- [5] PONTES, A. A. A. Uma Arquitetura de Agentes para Suporte à Colaboração na Aprendizagem Baseada em Problemas em Ambientes Virtuais de Aprendizagem. 2010. 114 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual do Rio Grande do Norte, Universidade Federal Rural do Semi-Árido, Mossoró, 2010.
- [6] RUSSELL, S. e NORVING, P. Inteligência Artificial. 2ª Edição. Rio de Janeiro: Campus, 2004.
- [7] BELLIFEMINE, F. L., CAIRE, G., GREENWOOD, D. Developing multi-agent systems with JADE. v. 5, Wiley, 2007.
- [8] SAVERY, J. R. Overview of Problem-based Learning: Definitions and Distinctions. The Interdisciplinary Journal of Problem-based Learning, v. 1, n. 1, p. 9-20, 2006.
- [9] VERONESE, G.; CORREA, A.; WERNER, C.; JEZINI NETTO, F. ARES: Uma Ferramenta de Engenharia Reversa Java-UML.In: SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SOFTWARE SBES, 16., Sessão de Ferramentas. Gramado: p. 347-352, 2002.
- [10] GOMES, A. K.; BERNADINI, F. C.; MONARD, M. C.; BATISTA, G. E. A. P. A. Uma Sintaxe Padrão Prolog para Classificadores Simbólicos. São Carlos: n. 154, 2002.
- [11] GRUBER, T. R. A translation approach to portable ontologies. Knowledge Acquisition, v. 5, n. 2, p. 199-220, 1993.
- [12] BECHHOFER, S.; HARMELEN, F. V.; HENDLER, J.; HORROCKS, L.; MCGUINNESS, D. L.; PATEL-SCHNEIDER, P. F.; STEIN, L. A. OWL Web Ontology Language Reference. W3C Recommendation, v. 10, [s.n.], 2004.
- [13] KNUBLAUCH, H.; FERGERSON, R. W.; NOY, N. F.; MUSEN, M. A. The Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications. Lecture Notes in Computer Science, Springer, v. 3298/2004: [s.n.], p. 229-243, 2004.
- [14] BRÓLIO, M.; OMAR, N.; FRANGO, I.; PIMENTEL, E. Modelagem de um Ambiente de Apoio à Avaliação Continuada Construído sob Ontologia. In: SÁNCHEZ, J. (Ed.). Nuevas ideas en Informática Educativa, v. 2, p. 64-71, 2006.
- [15] VIZCAÍNO, A. A Simulated Student Can Improve Collaborative Learning. International Journal of Artificial Intelligence in Education, v. 15, n. 1, p. 3-40, 2005.

- [16] MOISIL, I.; PAH, I.; BARBAT, B. POPA, E. M. Socio-cultural modelling of the student as the main actor of a virtual learning environment. In: PROCEEDINGS OF THE WSEAS, INTERNATIONAL CONFERENCE ON MATHEMATICAL METHODS AND COMPUTATIONAL TECHNIQUES IN ELECTRICAL ENGINEERING, 8., Bucharest: [s.n.], 2006.
- [17] LIMA, M. R. C.; LABIDI, S.; BASTOS FILHO, O. C.; FONSECA, L. C. C. Aprendizagem cooperativa e o problema de formação de grupos. Renote Novas Tecnologias na Educação, v. 3, n. 1, 2005.
- [18] SILVEIRA, S. R.; BARONE, D. A. C. Formação de grupos colaborativos em cursos a distância via web: um estudo de caso utilizando técnicas de inteligência artificial. Revista Brasileira de Informática na Educação, v. 14, n. 2, 2009.
- [19] FELIX, Z. C.; TEDESCO, P. A. Smart Chat Group: Ferramenta Ciente de Contexto para Formação de Grupos-Versão Final. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO SBIE, 19., Fortaleza: 2008.
- [20] DUEZ-RODRIGUEZ, H., MORALES-LUNA, G., e OLMEDO-AGUIRRE, J. O. Ontology-based knowledge retrieval. In: ARTIFICIAL INTELLINGENCE, 2008. MICAI'08. Seventh Mexican International Conference on, IEEE, p. 23–28, 2008.
- [21] GAVA, T. B. S. e MENEZES, C. S. Uma ontologia de domínio para a aprendizagem cooperativa. In: ANAIS DO SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, v. 1, [s.n.], p. 336–345, 2003.
- [22] TSENG, K., CHIANG, F., e HSU, W. Interactive processes and learning attitudes in a web-based problem-based learning (PBL) platform. Computers in Human Behavior, 24(3):940–955, 2008.
- [23] STROBEL, J. e VAN BARNEVELD, A. When is PBLMore Effective? AMeta-synthesis of Meta-analyses Comparing PBL to Conventional Classrooms. Interdisciplinary Journal of Problem-based Learning, 3(1):4, 2009.
- [24] SENDAG, S. e FERHAN ODABAS, H. Effects of an online problem based learning course on content knowledge acquisition and critical thinking skills. Computers & Education, 2009.