

Integração de soluções de Sequenciamento Curricular Adaptativo ao Moodle

João P. R. P. da Silva^{1,2}, Eduardo Barrere^{1,2}, Jairo Francisco de Souza^{1,2}

¹LApIC Research Group - Instituto de Ciências Exatas
Universidade Federal de Juiz de Fora
36.036-900 – Juiz de Fora – MG – Brasil

²Programa de Pós Graduação em Ciência da Computação
Universidade Federal de Juiz de Fora
36.036-900 – Juiz de Fora – MG – Brasil

{jpradd, eduardo.barrere, jairo.souza}@ice.ufjf.br

Abstract. *Intelligent and Adaptive Systems contribute to learning and facilitating the work of teachers and tutors. Approaches for recommending learning objects, for the most part, are not available for use in popular LMS such as Moodle. This work presents a proposal for retrieving and processing student data to feed an Adaptive Curriculum Sequencing (ACS) system. A plugin for Moodle was developed that uses an evolutionary approach to sequencing considering the course's characteristics, the students, and the available materials. The proposal was evaluated using a proof of concept.*

Resumo. *Os Sistemas Inteligentes e Adaptativos contribuem com o aprendizado e facilitam o trabalho dos professores e tutores. As abordagens para recomendação de objetos de aprendizagem, em sua maioria, não estão disponíveis para uso em LMS populares, como o Moodle. Este trabalho apresenta uma proposta de recuperação e tratamento de dados dos alunos para alimentar um sistema de Sequenciamento Curricular Adaptativo (SCA). Foi desenvolvido um plugin para o Moodle que utiliza uma abordagem evolutiva para sequenciamento considerando características do curso, do aluno e dos materiais disponíveis. A proposta foi avaliada utilizando uma prova de conceito.*

1. Introdução

O interesse em aprendizagem online com auxílio das ferramentas de ensino foi intensificado nos tempos atuais, principalmente após a pandemia do Covid-19 [Hong et al. 2022]. O mapeamento sistemático de [Kabudi et al. 2021] apresenta trabalhos de Inteligência Artificial (IA) voltados para a aprendizagem adaptativa (AA), com o intuito de auxiliar os alunos e professores. No entanto, a maioria dos trabalhos permanecem em estado experimental, sem perspectiva sólida de serem gerados produtos funcionais para a comunidade.

A AA possui eficácia comprovada [Alqahtani et al. 2021] e dentre as formas de adaptar a aprendizagem, está a criação de caminhos de aprendizado e recomendação de objetos de aprendizagem (OAs) adequados para cada aluno. Existe heterogeneidade das soluções para recomendação, mas poucas soluções estão disponíveis para uso em ambientes reais, que contam com diferentes tecnologias de LMS (*Learning Management Systems* ou Sistemas de Gestão da Aprendizagem) [Morze et al. 2021, Alqahtani et al. 2021].

Uma solução que faça uma interface entre sistemas de Sequenciamento Curricular Adaptativo (SCA) e LMS pode facilitar pesquisas sobre o impacto dessas tecnologias no ensino remoto, principalmente para pesquisadores que não são da área tecnológica.

Neste trabalho é apresentada uma proposta de *middleware* entre LMS e sistemas SCA, que conta com a recuperação de informações de conhecimento prévio dos alunos, tempo de estudo e perfil de preferência de aprendizagem. Essa solução abre caminho para o desenvolvimento de outras soluções de ensino adaptativo voltado a recomendação de OAs para LMS comuns. Isso possibilita a aplicação de mais ferramentas que estavam somente em fase experimental, como destacado em [Kabudi et al. 2021].

Os SCA e recomendadores de OAs tratam uma alta gama de materiais, reduzindo essa sobrecarga e dando qualidade ao curso [Yang et al. 2021]. Baseado no *Design Science Research* (DSR), foram gerados o modelo e o artefato propostos. O modelo de DSR adotado conta com investigação do estado da arte, levantamento das tecnologias semelhantes e elicitação dos requisitos do *software* científico. O artefato resultante é um *plugin* para o Moodle, trabalhando como *middleware* de um SCA de [Martins et al. 2021]. O artefato e o modelo foram validados por um experimento, testando seus módulos e comunicação, garantindo a viabilidade e contribuição para a comunidade.

O trabalho está estruturado da seguinte forma: a seção 2 apresenta uma fundamentação teórica sobre aprendizagem adaptativa relacionada a SCA. A seção 3 discute os trabalhos relacionados com a criação de sistemas de AA com recomendação de OAs. Em seguida, na seção 4 é apresentado o processo de avaliação com análise dos resultados. Na seção 5 apresentam-se as conclusões e trabalhos futuros.

2. Aprendizagem adaptativa com SCA

Com o avanço da IA voltada para sistemas de aprendizagem, surgiram ferramentas que possibilitam avanços para a AA [Kabudi et al. 2021]. A adaptação do ensino focada nos estudantes viabiliza aplicações de técnicas pedagógicas, que resultem em um maior sucesso no aprendizado. Na literatura, os trabalhos exploram os tipos de AA, preferência de estilos de aprendizagem e necessidades dos alunos, garantindo melhores resultados do que a ausência dessas adaptações [Alqahtani et al. 2021]. O uso de sistemas de recomendação para AA favorece uma maior eficácia do aprendizado *online* [Yang et al. 2021]. O mapeamento de [Marante et al. 2020], destaca o crescente número de trabalhos associando recomendação e AA, mas com muito ainda para ser explorado.

Uma alternativa proposta na literatura de AA trata os alunos de forma individual, com conteúdo personalizado [Martins et al. 2021]. O SCA é uma metodologia que atende bem essa função, seguindo as ideias de recomendação e propondo sequências de conteúdo adequadas para os alunos, reduzindo a desorientação e sobrecarga cognitiva. Uma das variáveis mais comuns há alguns anos para individualizar o conteúdo é através da identificação de um estilo de aprendizagem, o qual o uso do modelo *Felder and Silverman Learning Style Model* (FSLSM) [Felder 1988] foi muito popular. Nele os alunos são agrupados por perfis desse modelo e cada um possui níveis de afinidade variados em relação aos tipos de OAs. Atualmente é conhecido que esses estilos não possuem efeito na aprendizagem [Shamsuddin and Kaur 2020, Rogowsky et al. 2020], contudo, outros modelos de usuário tem sido usados no contexto de SCA para modelar preferências do estudante como forma de aumentar o engajamento e a motivação durante o estudo. Por

outro lado, basear a personalização do conteúdo somente pela preferência de materiais não garante um bom aprendizado, sendo preciso explorar outras características dos alunos como desempenho, por exemplo, [Marante et al. 2020]. Saber o nível de conhecimento prévio antes do sequenciamento, possibilita conteúdos que sejam adequados aos alunos [Dwivedi et al. 2018]. Outra variável importante a se considerar é o fator “tempo para o estudo”, visto que o SCA objetiva evitar uma sobrecarga ao estudo do aluno, viabilizando um conjunto de OAs adequado para assimilação do conteúdo.

3. Trabalhos relacionados

Nesta seção são analisados trabalhos sobre SCA e sistemas de recomendação de OAs. Para a eliciação dos pontos importantes dos trabalhos, foi observada a automatização e simplificação de informações obtidas dos professores e alunos. Foi analisado como os trabalhos trataram a variável do tempo do estudo dos alunos e o tratamento das estruturas de conhecimento e conceitos. Muitos sistemas adaptativos não são gratuitos ou acessíveis, sendo razoável ampliar opções portáteis e aplicáveis em outras LMS, como o Moodle [Morze et al. 2021, Kabudi et al. 2021]. Sendo assim, os trabalhos foram analisados se eles se adaptam a LMS e quanto à sua flexibilidade de uso de OAs de variadas fontes.

Para verificar a preferência de OAs pelos alunos, como parte das variáveis de recomendação, foi utilizado o modelo de Felder-Silverman em [Shchedrina et al. 2021], [Arsovic and Stefanovic 2020] e [Júnior et al. 2020]. O primeiro realizou como experimento um sistema adaptativo via *plugin* no Moodle, baseado no trabalho de [Limongelli et al. 2011]. Nessa abordagem, o professor deve ter um envolvimento maior na elaboração do curso, inserindo os conceitos, criando caminhos de aprendizagem e inserindo seus materiais. O ensino adaptativo ocorre utilizando informações sobre o conhecimento prévio do aluno via teste e utilização do método de Felder-Silverman para obter as preferências de aprendizado. Baseado nos caminhos e materiais disponibilizados pelo professor e os dados dos alunos, o algoritmo calcula, de tempos em tempos, se o aluno ampliou o conhecimento através de avaliações. Se obteve ganho de conhecimento, novos materiais são fornecidos para o aprendizado de um novo conceito.

A abordagem de [Arsovic and Stefanovic 2020] também cria um *plugin* para o Moodle. Na criação do curso, o professor escolhe e elabora os materiais para compor o acervo a ser recomendado. Como partida fria, realizam o teste para calcular o nível de conhecimento dos alunos e os agrupam com um modelo inspirado no método de FLSM. Para gerar a adaptação dos caminhos de aprendizagem com os OAs para cada aluno, o professor precisa realizar novos testes para atualizar os níveis de conhecimento.

O trabalho de [Júnior et al. 2020] também faz uso do método de FLSM e apresenta um algoritmo de recomendação bio-inspirado. A adaptação dos materiais ocorre com uso do nível cognitivo, objetivos e preferência de aprendizagem de cada estudante. O principal critério na seleção dos materiais é o uso da teoria *Set Coverage Problem*, escolhendo os materiais que possuem maior cobertura de assuntos. O sistema não está acoplado a um LMS, se apresenta como um protótipo e não fica claro como se obtém o nível de conhecimento dos alunos e outras variáveis. O professor informa o domínio de estudo, que serve para recuperação de materiais do YouTube e Wikipedia que contenham a maior cobertura dos assuntos desejados. O arquivamento de OAs segue o padrão IEEE-LOM + CLEO, visando facilitar o uso de outros repositórios do mesmo padrão.

Organizar e relacionar conceitos com apoio de estruturas de conhecimento como ontologias ajuda no processo de recomendação de materiais. [Melo et al. 2021] busca solucionar a recomendação de OAs de forma personalizada com um mapa de conhecimentos estruturados para organizar os conhecimentos a serem abordados pelos alunos. Após o professor inserir no sistema os conceitos para esse mapa, o sistema recomenda um conjunto de OAs (YouTube e Wikipedia) para o professor escolher, que tenham atingido uma boa cobertura sobre os assuntos. Os alunos são avaliados com um teste de conhecimento e por um teste de personalidade baseado em estudos anteriores dos mesmos autores. Com isso, o sistema recomenda um conjunto de materiais personalizados para cada aluno. O aluno deve percorrer o mapa de conhecimento e ser avaliado sempre que aprender um conceito. Após cada teste, o algoritmo de recomendação realiza uma nova análise, remove os nós aprendidos do mapa e inicia um novo ciclo de aprendizagem. Esse processo ocorre até que todos os conceitos sejam aprendidos. O sistema não foi desenvolvido para um LMS, mas sim como ferramenta independente.

Pensando em organização dos conceitos, [Silva et al. 2021] criaram uma plataforma própria para sugerir OAs de matemática e organiza os conceitos e categorias. Dentro desse sistema, o professor desenvolve seus próprios materiais e os relaciona a conceitos. Para gerar o modelo de aluno, o aluno realiza um teste de conhecimento e uma análise de perfil motivacional, baseada na sua autoconfiança e progresso (acertos) sobre os conteúdos. Dado o conteúdo baseado em níveis de dificuldade criado pelo professor e o modelo dos alunos, a ferramenta faz a recomendação dos próximos OAs aos alunos. A principal vantagem desse sistema se torna também sua desvantagem, pois é especializada em matemática, limitando o escopo de aplicação.

Já o trabalho de [Yang et al. 2021] apresenta grande flexibilidade no uso de repositórios no processo de recomendação. Os autores apresentam uma solução para AA baseada em um algoritmo de recomendação envolvendo *bigdata*. Os professores inicialmente definem os conceitos (metadados), estratégias de ensino e recursos de aprendizagem. Nesse estudo, os autores definiram uma base genérica como fonte de OAs. Para caracterizar o perfil do aluno, não existe uma aplicação inicial de teste de conhecimento, que só ocorre após o estudo dos primeiros materiais. No entanto, inicialmente, o aluno é questionado sobre seu estilo de aprendizagem ou, se preferir, respondendo um questionário fornecido. Os autores realizam registros de *log* sobre o tempo de acesso e uso, porém, não deixam claro se essas informações foram usadas apenas para o experimento ou se fazem uso disso no modelo do estudante. Os ciclos de adaptação são feitos a cada avaliação para refinar o perfil de aprendizagem e realizar ajustes para recomendar materiais aos alunos. Esse sistema não foi aplicado a nenhuma LMS.

Para melhor compreensão da cobertura de recursos dos sistemas de AA com recomendação de materiais e sequências curriculares, veja a Tabela 1, onde “✓” inclui o parâmetro, “\” parcialmente e “?” não é claro. Na última linha está incluído o presente trabalho como parâmetro de comparação.

Nos trabalhos analisados, o processo de obtenção de conhecimento prévio não foi automatizado, como, por exemplo, verificar o histórico de notas do aluno. A obtenção desses dados de forma manual exige elaboração de testes pelo professor e dedicação do aluno no início do curso. Outra opção seria um início sem teste, mas prejudicando o possível contato inicial do aluno com materiais pouco relevantes ou complexos para o

Tabela 1. Cobertura de recursos dos trabalhos

Trabalhos	1	2	3	4	5	6
[Melo et al. 2021]			✓	\	\	
[Shchedrina et al. 2021]			\		\	✓
[Silva et al. 2021]			✓		\	
[Júnior et al. 2020]			✓	✓	✓	
[Yang et al. 2021]		?	\	✓	\	
[Arsovic and Stefanovic 2020]				✓	\	✓
Presente Trabalho	✓	✓	✓	✓	✓	✓

1- Obtenção de conhecimento prévio automatizado; 2- Uso do tempo do aluno para a adaptação; 3- Estruturação e automatização no uso de conceitos; 4- Flexibilidade no uso de repositórios; 5- Automatização na escolha e recomendação dos materiais; e 6- Usado ou replicável em LMS.

estágio atual de aprendizagem, como em [Yang et al. 2021]. O presente trabalho oferece uma recuperação do conhecimento prévio do aluno de outros cursos do Moodle, melhorando a partida fria, sem exigir intervenção inicial do professor e do aluno. A consideração do tempo disponível de estudo do aluno também não é levado em conta pelos outros trabalhos. Na prática, os cursos possuem tempo para começar e terminar, sendo importante recomendar conteúdos que atendam o tempo de estudo do aluno. Este fator é considerado no algoritmo de SCA escolhido para esse trabalho.

A maioria dos trabalhos estruturou ou facilitou a inserção dos conceitos, flexibilizou o uso de repositórios ou automatizou a escolha de OAs. No entanto, alguns possuem pouca automatização para inserção de conceitos, que poderiam simplificar com apoio de ontologias, mapas conceituais e categorias, como em [Júnior et al. 2020], [Melo et al. 2021] e [Silva et al. 2021]. Quanto aos repositórios e escolhas de OAs, existe um problema comum nos trabalhos, onde boa parte do esforço de preparação, busca e criação de materiais ainda ficam a cargo do professor. Não é foco de um sistema adaptativo retirar do professor a autonomia de criação de conteúdos, mas é possível facilitar o docente ao recomendar OAs de excelentes repositórios e bases diretamente aos alunos. Essa recomendação automática e direta só foi explorada por alguns dos trabalhos.

Além do presente trabalho, somente outros dois buscaram flexibilizar o acesso, escolhendo um LMS popular. Devido a variedade de abordagens, existe uma complexidade para integrar e adaptar SCA a um LMS. Diante do conjunto de trabalhos relacionados, o *middleware* aqui apresentado possui uma cobertura ampla de pontos importantes e alguns pouco explorados no contexto de recomendação de materiais para ensino adaptativo.

4. Processo de avaliação e discussão

Com o objetivo de apresentar e validar o *middleware*, foi utilizado DSR, uma metodologia de construção de conhecimento científico, que envolve o desenvolvimento de um artefato inovador [Dresch et al. 2015]. Por isso, baseado na abordagem de [Nunamaker et al. 1990], foi realizada a pesquisa e exploração do tema para a elicitação e validação do artefato. O DSR é dividido em etapas sobre o processo de projetar soluções científicas. A primeira etapa, chamada de “Construir um *framework* conceitual”, requer a definição do problema e a exploração de funcionalidades, processos de construção e

ideias de trabalhos e sistemas semelhantes. Após essa etapa investigativa, o foco é “Desenvolver uma arquitetura de sistema”, ou seja, pensar na arquitetura, extensibilidade, modularidade, e as funcionalidades dos componentes e sua intercomunicação. A terceira etapa é “Analisar e projetar o sistema”, que consiste em explicar o que está sendo estudado e o conhecimento científico analisando possíveis soluções. Ainda nessa etapa, é definida a proposta de solução escolhida do sistema, garantindo a continuidade da pesquisa. A quarta etapa consiste em “Construir o protótipo do sistema”, onde o artefato é verificado quanto ao seu comportamento em situações próximas do real. Segundo os autores, essa etapa é essencial para verificar a viabilidade e possíveis problemas futuros. Com isso, o projeto pode ser ajustado se necessário e buscar responder as questões da pesquisa. A última etapa é “Observar e avaliar o sistema”, assim deve-se realizar estudos de caso, testes, validação e uso de métricas para medir qualidade e performance do artefato, quanto ao que foi determinado na primeira etapa. Ao final dessa etapa, pode ser feita a consolidação de propostas e modelos para apoiar outros pesquisadores.

As três seções anteriores respondem às necessidades da primeira etapa do *framework* conceitual, apresentando o problema, objetivos e estudos na literatura. Agora é apresentado o resultado do desenvolvimento das demais etapas do DSR, esclarecendo a arquitetura, funcionalidades, experimentos e discussão do artefato e modelo obtidos.

4.1. Arquitetura do sistema

A arquitetura é baseada em um *middleware*, que trabalha entre um LMS e um SCA. No presente trabalho, o *middleware* foi adaptado ao Moodle, sendo desenvolvido como um *plugin* para o experimento. O Moodle oferece um bom suporte para desenvolvimento de *plugins* com suporte para API Rest e acesso facilitado ao seu banco de dados MySQL. Para a criação do artefato, foi gerada a arquitetura apresentada na Figura 1.

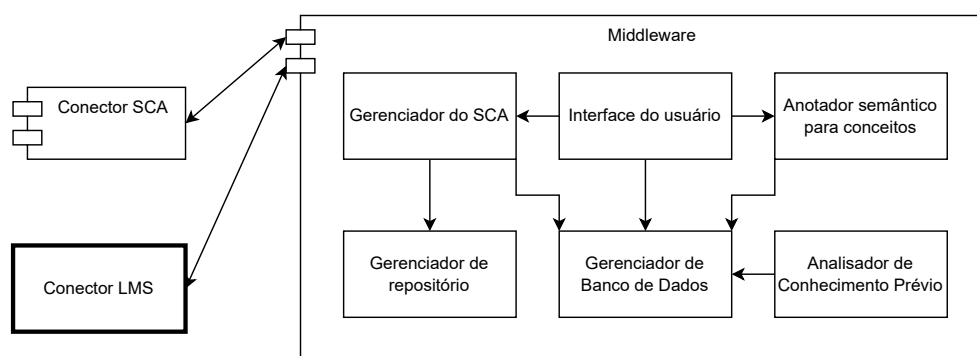


Figura 1. Arquitetura com os módulos do *middleware* e a intermediação do mesmo com o LMS e o SCA.

O *middleware* foi criado de forma modular, possibilitando uma fácil manutenção e adaptação para futuras pesquisas, sendo possível a substituição de módulos, como o “Anotador semântico para conceitos” (ASC) e “Analisador de Conhecimento Prévio” (ACP). Outras adaptações possíveis são em regras internas dos outros módulos, como do “Gerenciador de Banco de Dados” focado na LMS de escolha. O “Gerenciador de Repositório” faz a tarefa de comunicação com repositório de padrão LOM (*Learning Object Metadata*) e pode ser adaptado para outros contextos e padrões durante o reuso. A organização proposta possibilita a reaplicação do modelo para outros LMS.

O ACP recupera a informação das notas e conceitos associados em disciplinas anteriores. Nas disciplinas, é comum que a maioria dos conceitos sejam novos, porém, alguns conceitos podem ter sido aprendidos integralmente ou parcialmente em outras disciplinas. Essa análise ajuda a verificar se o aluno possui algum conhecimento prévio do assunto, a fim de recomendar materiais introdutórios ou mais avançados. Com a análise dos trabalhos relacionados, os autores citados realizaram testes para obter o conhecimento prévio dos alunos para partida fria dos sistemas de recomendação, mas nenhum autor explorou a recuperação da informação desse conhecimento por outros meios. O presente trabalho mostra uma nova abordagem para recuperar o desempenho dos alunos sem que o professor crie uma avaliação antes do começo do curso (ver Seção 4.2).

O módulo ASC recebe o texto de plano de disciplina do professor. Esse documento é comum nos cursos de instituição de ensino, servindo de guia para abordar os temas da disciplina e permitindo os alunos terem ciência do conteúdo programático. O único trabalho do professor será submeter esse documento e aprovar os conceitos propostos. Como o documento pode apresentar padrões dos mais variados, o anotador semântico poderá extrair as palavras chave do conteúdo da disciplina. O anotador conta com apoio da ontologia DBpedia e cálculos de similaridade. O anotador utilizado é o mesmo do *framework* de [Martins et al. 2021], que extrai metadados de videoaulas através de reconhecimento de fala. Os conceitos selecionados são apresentados ao professor para confirmar a seleção, sem tirar a liberdade de acrescentar outros.

O módulo “Gerenciador do SCA” é o núcleo do *middleware* e realiza todo o processo de preparação para a execução do SCA. Esse módulo organiza e envia em arquivos os dados e parâmetros necessários para o SCA. Os dados desse gerenciador advêm do “Gerenciador de repositório” que carrega a listagem dos OAs com seus metadados e do “Gerenciador de Banco de Dados”, que coleta as informações processadas pelos outros módulos e salvas no LMS. O SCA adotado gera os OAs recomendados, sendo um recurso externo ao *middleware* que faz a comunicação com o LMS. O “Gerenciador do SCA” recebe de volta os dados processados em tuplas de alunos e listagem de materiais para cada um. Depois a lista do material se torna visível aos alunos da disciplina.

A “Interface do usuário” é o módulo da camada mais próxima do usuário, permitindo a interação do professor e alunos. O perfil do professor oferece telas para gerir a submissão do plano, conferência dos conceitos e a chamada da execução do SCA. Nessa última tela, deve ser informada qual lista de conceitos deve ser utilizado e um nome para o ciclo de estudos. Além de uma interface do teste de preferência de aprendizagem, o aluno visualiza por outra tela a listagens de OAs que deve estudar e gerenciar seu estudo. A “Interface do usuário” varia de acordo com as informações de entrada para cada SCA adotado e técnicas de interação humano-computador voltadas ao ensino. Na próxima seção tem mais detalhes sobre a abordagem desse trabalho sobre a interface.

4.2. Exemplo de funcionamento do sistema

Para analisar e projetar o *middleware*, este foi adaptado para o Moodle e um SCA de código aberto¹. Dessa maneira, o *middleware* foi projetado como um *plugin*. Essa seção descreve o funcionamento prático do sistema dentro desse ambiente virtual de aprendizagem, passando por todos os módulos já apresentados. Após a criação do curso dentro

¹Disponível em ²

do LMS, o professor tem acesso ao *plugin* criando, na listagem do curso, o artefato como uma atividade. Dessa forma, a atividade poderá ser acessada pelos alunos e professor. Inicialmente, os alunos acessam o recurso para gerar sua preferência de aprendizagem e dizer sua pretensão de quantidade de horas para dedicar a disciplina. O professor submete o plano de atividades da disciplina, do qual será retornado do ASC uma lista de conceitos para serem selecionados. O SCA adotado pode ser utilizado para um ciclo único (gerando os OAs para o curso todo) ou pode ser executado para cada conjunto de conceitos criados pelo professor após ciclos de avaliações do curso (o mais comum entre os SCA). Isso permite flexibilidade extra ao professor ao gerar a adaptação do aprendizado. Outro ponto que é específico do SCA adotado, é o preenchimento da informação do tempo esperado de estudo para os alunos, caso eles não tenham informado previamente (Figura 2).

Conceitos desejados para a nova etapa *	Plano 1 ▼
Tempo de estudo (Caso aluno não tenha definido) *	<input type="text"/> minutos
Nome do ciclo de estudo *	<input type="text"/>

Gerar

Figura 2. Tela de configuração e chamada do SCA (exemplo aplicado no Moodle).

Submetido essas informações, o módulo ACP explora o histórico de disciplinas do Moodle cursadas pelos alunos. Esse módulo busca similaridade entre os conceitos já aprendidos anteriormente para saber se os alunos possuem já alguma base sobre os assuntos que serão abordados no curso. O conhecimento prévio está nivelado na escala de Likert, onde 1 é nenhum conhecimento prévio sobre o assunto e 5 é caso o aluno já tenha aprendido o conceito anteriormente com nota máxima. Esse valor é calculado sobre a similaridade entre categorias da DBpedia que representam esses conceitos e as notas obtidas nas disciplinas. Essa informação de conhecimento prévio é pré-requisito para o SCA, como na maioria dos trabalhos relacionados.

O professor requisita o processamento sobre o plano que escolheu, tempo esperado de estudo e nomeia o ciclo de estudo, como ilustrado na Figura 2. O ciclo é uma possibilidade do professor utilizar o SCA várias vezes em vez de uma recomendação única para todo o curso. Então, o “Gerenciador do SCA” já possui do banco do Moodle todos os dados necessários para gerar a recomendação dos OAs aos alunos. No caso do SCA em questão, os dados de entrada são: os conceitos, dados de conhecimento prévio, tempo para estudo e preferência de aprendizagem dos alunos e listas de OAs com seus metadados. Esse gerenciador organiza todos os arquivos de entrada do SCA e recebe o arquivo com a sequência de materiais para cada aluno. Esses dados ficam visíveis para o

professor e para o aluno. O aluno pode, para cada material, marcar se já estudou, saber o tempo estimado de estudo, marcar sua satisfação com o mesmo de 1-5 estrelas, saber o tempo total de estudo para todos os materiais e escrever um *feedback* para o professor (na Figura 3 mostra um OA como exemplo). O módulo focado na interface de usuário é importante por facilitar o acesso aos dados e comunicação aluno-professor. Este é um módulo que deve ser adaptado dependendo da LMS a ser utilizada. No entanto, no presente trabalho, foram construídas as telas de *plugin* aproveitando os recursos nativos do Moodle, mas com ideias conceituais para servirem de modelo para outros LMS.

Objeto de aprendizagem	Visto	Link	Tempo	Satisfação	Tempo Total	Feedback (opcional)
Algoritmos e Representações		Link 0001	3:41	★★★★★	26:03	<input type="text"/>

Figura 3. Recorte de tela de um OA direcionado ao aluno e as opções de *feedback* e organização pessoal.

4.3. Viabilidade do projeto

[Dresch et al. 2015] destaca que, no modelo DSR proposto por [Nunamaker et al. 1990], a construção do protótipo é fundamental para verificar a sua viabilidade em um contexto real ou próximo do real. Baseado nos resultados, pode ser afirmado quais são as necessidades de ajuste e garantir se o sistema atenderá a questão de pesquisa. Para tal, via protótipo, foram inseridos dados para serem tratados no fluxo de processamento do *middleware*, entre os módulos e analisar os resultados. O protótipo foi instanciado para o Moodle e para a solução de SCA citada na subseção anterior. Para os dados da turma, foram inseridos 25 alunos com preferências de aprendizagem geradas à partir do FSLSM. O *dataset*³ é o mesmo utilizado em outros trabalhos para testar soluções de SCA e foi rotulado com estilos de aprendizado. Neste experimento, os estilos são utilizados apenas como modelo de preferência do usuário, o que irá influenciar na prioridade de escolha do material, mas não como restrição do conjunto de materiais candidatos do SCA.

O plano de ensino foi organizado para abranger tópicos da área de Ciência da Computação para iniciantes da área, dando uma visão geral, mas superficial, dos assuntos principais da área. Deste plano, o ASC extraiu 30 conceitos, dentre eles “Big Data”, “Paradigmas de Programação” e “Protocolos de Comunicação”, por exemplo. O repositório adotado conta com 284 OAs da área da computação no padrão LOM, que permitiu saber a cobertura dos conceitos tratados nos materiais. O histórico dos alunos foi gerado de forma aleatória e foram registrados como aprovados em uma ou mais disciplinas, a saber: “Algoritmos”, “Rede de Computadores”, “Banco de Dados”, “Engenharia de Software”, “História da Computação”, “Sistemas Numéricos e Lógica”, “Sistemas Operacionais” e “Organização de Computadores”. Com isso, o ACP gerou uma pontuação do nível de conhecimento prévio dos alunos para cada um dos 30 conceitos baseados na similaridade dos assuntos relacionados às disciplinas já cursadas.

O “Gerenciador do SCA” realizou as consultas e geração dos documentos necessários para o processamento do SCA. O módulo fez a comunicação via *webservice* e

³<https://github.com/martinsadw/evolutionary-computation/tree/master/instances/real/LOM>

tratou o retorno dos dados, coletando o valor das tuplas aluno-sequência. Os dados foram salvos no Moodle e acessíveis pela interface. O número médio de OAs por aluno foi de 98,48, com desvio padrão de 6,75. A eficácia da recomendação dos OAs está atrelada ao método do SCA e não é objeto dessa análise. O objetivo desse estudo, contudo, é verificar se o *middleware* é capaz de ser adaptado para uma instância de LMS e SCA.

O protótipo dessa forma se mostrou viável por atender os requisitos e funcionalidades desejadas para integrar soluções genéricas de SCA em LMS. No dado momento, foi concluída a quarta etapa de 5 do DSR, que garante a viabilidade do projeto. A última etapa do DSR exige um estudo de caso e avaliação do sistema quanto usabilidade e satisfação dos usuários. O *software* em questão mostra que aplicar SCA em LMS é viável por essa arquitetura demonstrada. A adaptação curricular é pouco explorada nos LMS mais populares segundo [Morze et al. 2021]. Muitos projetos de recomendação de OAs são limitados a experimentos e não são projetados para uso comum das instituições de ensino.

5. Conclusões

Houve um grande aumento nos últimos anos de ferramentas para AA e voltadas para recomendação de OAs e caminhos de aprendizagem. No entanto, poucas foram as ferramentas realmente disponibilizadas, normalmente estando em estado experimental e sem serem aplicadas em LMS populares [Morze et al. 2021, Kabudi et al. 2021]. Alguns trabalhos foram analisados e, além da baixa portabilidade dos modelos, muitos não se preocuparam em facilitar o trabalho dos professores, que precisam intervir em vários momentos novos, como selecionar OAs ou criar um pré-teste antes do início do curso. Os sistemas também não pensaram como os cursos funcionam na vida real, como no caso do tempo limitado que os alunos possuem para estudar e concluir as disciplinas.

Diante desses problemas, este trabalho, baseado na metodologia do DSR, realizou essa análise da literatura sobre trabalhos mais recentes da área para levantamento das fraquezas de outros sistemas e para uma elicitación de um artefato científico. Com isso, foi criada uma arquitetura de *middleware* para conectar ferramentas SCA com um LMS. No presente trabalho, o *middleware* foi projetado como um *plugin* no Moodle, para dar conectividade a um SCA. O sequenciador adotado possui como parâmetros o perfil de preferência de aprendizado, os conceitos da disciplina, as OAs com os seus metadados e o nível de conhecimento dos alunos. Estes parâmetros são os mais comuns para esse tipo de sistema adaptativo. Um fator extra, que também é parâmetro desse SCA, é a variável tempo, para não gerar sequências de conteúdo que deixem o estudante sobrecarregado.

Este trabalho contribui com um artefato validado sobre sua viabilidade. Foi desenvolvido um módulo de recuperação do conhecimento prévio dos alunos baseado no seu histórico de disciplinas do Moodle. Desta forma, o conteúdo previamente estudado é relacionado com conceitos, sem necessidade de aplicação de pré-testes. O modelo aqui apresentado pode ser reaproveitado e adaptado para outros SCA ou com módulos de outros pesquisadores. Foi verificada a validade do *middleware* com o Moodle. Embora o Moodle seja um LMS popular, existem diversos LMS e são necessárias investigações com esses outros sistemas. Ainda, é necessário esforço de investigação com uma base de usuários que represente os diferentes atores do sistema. Quanto a última etapa do DSR, que envolve um estudo de caso, será foco de um trabalho futuro.

Referências

- Alqahtani, R., Kaliappen, N., and Alqahtani, M. (2021). A REVIEW OF THE QUALITY OF ADAPTIVE LEARNING TOOLS OVER NON-ADAPTIVE LEARNING TOOLS. *International Journal for Quality Research*, 15(1):45. Publisher: University of Kragujevac, Faculty of Engineering.
- Arsovic, B. and Stefanovic, N. (2020). E-learning based on the adaptive learning model: case study in Serbia. *Sādhanā*, 45(1):266.
- Dresch, A., Lacerda, D. P., and Antunes, J. A. V. A. J. J. (2015). *Design Science Research: Método de Pesquisa para Avanço da Ciência e Tecnologia*. Bookman Editora. Google-Books-ID: M63XDwAAQBAJ.
- Dwivedi, P., Kant, V., and Bharadwaj, K. K. (2018). Learning path recommendation based on modified variable length genetic algorithm. *Education and Information Technologies*, 23(2):819–836.
- Felder, R. M. (1988). LEARNING AND TEACHING STYLES IN ENGINEERING EDUCATION. *Eng Educ*, page 11.
- Hong, S., Kwon, H.-G., and Han, H. (2022). An Exploratory Study on Online Education Scaffolding Strategy and Effect at the University Using a Systematic Review. *Journal of Digital Convergence*, 20(4):63–72. Publisher: The Society of Digital Policy and Management.
- Júnior, C. P., Júnior, C. F. B., Araújo, R. D., and Dorça, F. A. (2020). Personalized Recommendation of Learning Objects Through Bio-inspired Algorithms and Semantic Web Technologies: an Experimental Analysis. In *Anais do Simpósio Brasileiro de Informática na Educação*, pages 1333–1342. SBC. ISSN: 0000-0000.
- Kabudi, T., Pappas, I., and Olsen, D. H. (2021). AI-enabled adaptive learning systems: A systematic mapping of the literature. *Computers and Education: Artificial Intelligence*, 2:100017.
- Limongelli, C., Sciarrone, F., and Giulia Vaste (2011). Personalized e-learning in Moodle: the Moodle.Ls System - Learning & Technology Library (LearnTechLib).
- Marante, Y., Silva, V. A. A. d., Jr, J. G., Vitor, M. A., Martins, A. F., and Souza, J. F. D. (2020). Evaluating Educational Recommendation Systems: a systematic mapping. In *Anais do Simpósio Brasileiro de Informática na Educação*, pages 912–921. SBC. ISSN: 0000-0000.
- Martins, A. F., Machado, M., Bernardino, H. S., and de Souza, J. F. (2021). A comparative analysis of metaheuristics applied to adaptive curriculum sequencing. *Soft Computing*, 25(16):11019–11034.
- Melo, S. L. d., Sousa, R. M. d. S., and Lima, L. V. (2021). Arquitetura de um sistema tutor inteligente para recomendação personalizada de objetos de aprendizagem considerando a teoria dos mapas de conhecimentos estruturados. *Research, Society and Development*, 10(16):e518101623831–e518101623831. Number: 16.
- Morze, N., Varchenko-Trotsenko, L., Terletska, T., and Smyrnova-Trybulska, E. (2021). Implementation of adaptive learning at higher education institutions by means of Mo-

- odle LMS. *Journal of Physics: Conference Series*, 1840(1):012062. Publisher: IOP Publishing.
- Nunamaker, J. F., Chen, M., and Purdin, T. D. (1990). Systems Development in Information Systems Research. *Journal of Management Information Systems*, 7(3):89–106. Publisher: Routledge _eprint: <https://doi.org/10.1080/07421222.1990.11517898>.
- Rogowsky, B. A., Calhoun, B. M., and Tallal, P. (2020). Providing instruction based on students' learning style preferences does not improve learning. *Frontiers in Psychology*, 11:164.
- Shamsuddin, N. and Kaur, J. (2020). Students' learning style and its effect on blended learning, does it matter?. *International Journal of Evaluation and Research in Education*, 9(1):195–202.
- Shchedrina, E., Valiev, I., Sabirova, F., and Babaskin, D. (2021). Providing Adaptivity in Moodle LMS Courses. *International Journal of Emerging Technologies in Learning (iJET)*, 16(02):95–107. Number: 02.
- Silva, V. A., Ferreira, H. N. M., Torres, A. d. A., and Rodrigues, F. C. (2021). Math Suggestion: Uma Ferramenta de Recomendação de Objetos de Aprendizagem Fundamentada nos Princípios das Avaliações de Autoeficácia e Análise de Desempenho. In *Anais do Simpósio Brasileiro de Informática na Educação*, pages 237–248. SBC. ISSN: 0000-0000.
- Yang, Y.-z., Zhong, Y., and Woźniak, M. (2021). Improvement of Adaptive Learning Service Recommendation Algorithm Based on Big Data. *Mobile Networks and Applications*, 26(5):2176–2187.