

Recomendação Híbrida de Objetos de Aprendizagem a partir da Predição de Necessidades Personalizadas do Estudante

Lucca Alexandre Schrammel¹, Lisandra Manzoni Fontoura¹

¹ Programa de Pós-Graduação em Ciência da Computação – Universidade Federal de Santa Maria (UFSM)

lucca.a.s@hotmail.com, lisandra@inf.ufsm.br

Abstract. *Given the large amount of data available, it is increasingly challenging to identify learning objects (LO) that will contribute to the student's learning process. In this scenario, this work proposes a hybrid model framework for LO recommendation based on the needs of each student. For this, the framework carries out a performance prediction process to identify difficulties that a student may present, and the framework generates assertions, increasing the chances of approval, considering that it will have a range of LO that will allow a better performance within an academic semester.*

Resumo. *Dada a grande quantidade de dados disponíveis, está cada vez mais difícil identificar objetos de aprendizagem (OA) que contribuirão no processo de aprendizagem dos estudantes. Neste cenário, este trabalho propõe um framework de modelo híbrido para recomendação de OA com base nas preferências e nas necessidades de cada estudante. Para isso, o framework realiza um processo de predição de desempenho para identificar eventuais dificuldades que um estudante poderá apresentar e o framework gera recomendações assertivas, aumentando as chances de aprovação, tendo em vista que o mesmo contará com uma gama de OA que permitirão um melhor desempenho dentro de um semestre letivo.*

1. Introdução

Sistemas de recomendação (SR) são popularmente conhecidos como uma ferramenta útil para auxiliar os estudantes a encontrar informações de interesse pessoal, o que pode ser utilizado para resolver a sobrecarga de informações que os alunos recebem durante a realização de algum curso ou disciplina específica. SR também podem filtrar aquilo que é de maior interesse do estudante, auxiliando-o a ter uma jornada de aprendizado mais ampla em relação ao que é apresentado em sala de aula [Pinho *et al.* 2019]. Deste modo, SR podem ser implementados em ambientes virtuais de aprendizagem e ferramentas colaborativas [Maria *et al.* 2019].

É possível usar SR em várias situações no ensino, tais como: personalizar o processo de ensino-aprendizagem de acordo com o perfil de aprendizagem do aluno [Barreto *et al.* 2014], adaptar a experiência do aluno visando o aumento da satisfação ajudar estudantes a escolher matérias ou cursos [Aguiar *et al.* 2018], recomendar faculdades e universidade [Moraes *et al.* 2018].

Devido um aumento significativo de recursos educacionais disponíveis para os estudantes [Marques 2022], as instituições de ensino estão cada vez mais buscando recursos digitais para utilização por parte dos estudantes. Essa situação pode gerar uma sobrecarga de dados a serem processados pelos estudantes, o que faz com que os SR possuam um papel importante no auxílio aos estudantes, tendo em vista que estes podem facilmente apoiar a tomada de decisão e preencher lacunas na aprendizagem dos alunos, com base na recomendação personalizada e nas necessidades individuais [CAMPOS *et al.* 2017]. Além disso, não foram evidenciados SR que levam em consideração o perfil do estudante e um provável desempenho que possa ser otimizado mediante a recomendação de OA, avaliando não somente o cenário atual, mas um cenário futuro que pode ocorrer com base nos padrões de aprendizagem do estudante.

Este trabalho propõe a criação de um *framework* de modelo híbrido para recomendação de materiais didáticos para estudantes matriculados em uma disciplina de algum curso. O foco é fazer com que os estudantes possam receber recomendação de materiais didáticos que fará com que as chances de êxito e aprovação sejam otimizadas, uma vez que o processo de recomendação considerará as preferências e as necessidades do estudante durante o processo de aprendizagem de uma determinada disciplina, antecipando eventuais dificuldades.

A proposta visa contribuir para a evolução dos SR na educação, partindo não apenas das preferências do estudante, mas incrementando critérios de necessidades e deficiências dos estudantes para a geração de recomendações mais precisas e personalizadas, com base no perfil de cada estudante. Esta motivação ocorre, pois, segundo estudo realizado por Silva (2021), 50% dos SR na educação partem de uma abordagem tradicional, ou seja, as recomendações são geradas apenas com base nas preferências dos estudantes, o que pode desconsiderar suas reais necessidades.

O artigo está organizado da seguinte forma: na Seção 2 são descritos alguns conceitos necessários para o entendimento do trabalho, na Seção 3 são descritos trabalhos relacionados e uma comparação com este trabalho, na Seção 4 é explicada a metodologia seguida para desenvolvimento do trabalho, na Seção 5 é detalhada a validação realizada e na Seção 6 o trabalho é concluído e são descritos trabalhos futuros.

2. Referencial Teórico

Sistemas de recomendação são ferramentas que visam sugerir informações para os usuários de forma personalizada, com base em seus interesses e necessidades, porém nem todos são iguais, existem vários sistemas e com uma gama considerável de formas de geração destas recomendações.

A filtragem colaborativa visa recomendar ao usuário itens que outros utilizadores com interesses semelhantes [Bobadilla *et al.* 2013]. Para Goldberg *et al.* (1992), este modelo de recomendação assume a hipótese de que usuários com um modelo de interações no passado terão uma tendência a manter o mesmo comportamento no futuro. Segundo Lops *et al.* (2011), os SR que são baseados em filtragem de conteúdo assumem como característica predominante a utilização de metadados dos itens combinados com o histórico de interações dos utilizadores. Esta técnica de recomendação assume a hipótese de que utilizadores possuem uma tendência a consumir itens que possuem características semelhantes ao que o próprio utilizador já

consumiu no passado [Thorat *et al.* 2015]. Filtragem baseada em conteúdo busca entregar sugestões ao utilizador com base no conhecimento da necessidade de um determinado item. Alguns autores acabam denominando este modelo de filtragem como “Recomendação Utilitária” [Guttman *et al.* 1998].

Combinando uma série de modelos de filtragem para SR, a filtragem híbrida é um modelo com grande destaque devido aos bons resultados alcançados e a grande capacidade de adaptação [Bobadilla *et al.* 2013, Balabanovic *et al.* 1997]. Moraes *et al.* (2018) destaca que as abordagens híbridas vêm ganhando atenção e se mostrando consistentes para recomendação de recursos educacionais, devido a sua natureza de criar um método mais robusto e com melhores resultados.

Para Cazella *et al.* (2012), os SR educacionais trabalham como filtros de informações, encaminhando o melhor conteúdo e informação que atende o aluno frente às suas necessidades. O uso crescente dos sistemas *e-learning* tornaram a coleta de dados mais fácil e possibilitaram fornecer aos aprendizes conteúdos educacionais individualizados [Drachsler *et al.* 2015].

A diferença encontrada entre alunos durante o processo de aprendizagem e nos resultados alcançados em pesquisas no estado da arte, conduziu ao entendimento da existência de diferentes estilos de aprendizagem que tem influência no aprendizado e de como cada indivíduo recebe e compreende as informações de forma a transformá-la em conhecimento [Kolb 1984]. Uma das maiores contribuições de Felder e Silverman foi a criação de um padrão que explica como um aluno comporta-se e relaciona-se diante da informação. Com base em pesquisas de Kolb (1984), que foca na aprendizagem experimental e em estudos de Jung (1971), que apresenta como as funções psicológicas compreendem e se orientam com o mundo externo, foi criado o Modelo de Estilo de Aprendizagem Felder-Silverman [Felder e Henriques 1995]. Este modelo é composto por quatro dimensões, que são: processamento (ativo e reflexivo), determina como as informações são percebidas e convertidas em conhecimento; percepção (sensorial e intuitivo), determina como as pessoas percebem o ambiente ao seu redor; entrada (visual e verbal), determina como os alunos preferem receber o conhecimento; organização (sequencial e global), determina a forma que se acompanha um assunto.

3. Trabalhos Relacionados

Nessa seção são descritos alguns trabalhos relacionados que foram utilizados para embasar o desenvolvimento da pesquisa em questão, utilizando conceitos e aplicações já existentes para a elaboração do *framework* proposto.

Imran *et al.* (2016) propõem um SR chamado PLORS. PLORS realiza a recomendação de diversos OA, tais como vídeos, atividades práticas, aplicações na vida real, e exemplos de aplicações a partir dos objetos que o próprio usuário visitou ou que outros estudantes com perfis semelhantes visitaram. Para se obter as informações de histórico de utilização e comportamento do usuário, são verificados os logs de acesso e utilização do sistema que contém a plataforma do curso. O sistema foi utilizado dentro de um curso, mas a eficácia das recomendações não foi mencionada.

Sweeney *et al.* (2016) desenvolveram um SR que auxilia o aluno na seleção da grade de disciplinas a serem cursadas em um curso de graduação, a partir do desempenho atual e já conhecido deste mesmo estudante. Os autores buscam técnicas de recomendação já utilizadas em *e-commerce* para análise do progresso do aluno e para

embasar as recomendações realizadas, visando um fluxo de aprovação contínua do estudante seguindo uma ordem cronológica de disciplinas viáveis a partir do seu desempenho. Este sistema realiza um processo de simulação das notas dos alunos nas disciplinas em que eles se matricularão no próximo período por meio de padrões de aprendizagem com base nos dados históricos de cada um dos estudantes.

Santos (2020) propõe solucionar o problema da recomendação de atividades aos alunos do Moodle, por meio do uso de um conjunto de dados educacionais, tendo como finalidade que essas recomendações possam ajudar o usuário a sempre encontrar questões que possibilitem maximizar o seu aprendizado. O sistema realiza a coleta de informações de acertos e erros em questões para geração de uma análise do perfil do estudante e então a recomendação é gerada a partir da filtragem colaborativa.

Hoic-Bozic *et al.* (2016) propuseram o modelo *Blended Learning* que une um sistema de gestão de aprendizagem, ferramentas *Web* e o *E-Learning Activities Recommender System* (ELARS) para melhorar a aprendizagem *online* personalizada. Foram implementadas as técnicas de recomendação baseadas em conhecimento, baseadas em conteúdo e/ou filtragem colaborativa, adaptados ao domínio educacional.

Carvalho *et al.* (2014) buscaram realizar o vínculo de OA com o modelo de Felder-Silvermann, que aborda os diferentes estilos de aprendizagem. Para isso, construíram um protótipo para gerar uma pontuação que aproxima ou distancia os OA para cada estilo de aprendizagem conhecido. Esta distância é medida com base na análise de uma série de metadados que cada OA possui, dessa forma, é gerada uma lista ordenada com os objetos mais próximos de cada um dos estilos de aprendizagem.

Este trabalho tem como diferenciais o fato que o processo de recomendação parte das preferências do estudante e leva em consideração as deficiências no aprendizado que cada aluno tem encontrado no decorrer da execução de uma determinada disciplina. Com base nessas deficiências, é criado um fluxo de recomendação vinculado ao plano de ensino do professor. Desta forma, as recomendações são diretamente atreladas aos objetivos que o estudante possui em cada uma das disciplinas. Acredita-se que com as recomendações devidas o estudante pode otimizar o seu desempenho em tempo hábil para uma aprovação no período letivo atual, para então, realizar uma evolução ao natural.

4. Metodologia

O presente trabalho tem como objetivo a implementação de um *framework* capaz de realizar recomendação de OA para estudantes de uma forma geral, não apenas levando em consideração suas preferências e características, mas também as suas dificuldades de aprendizagem. Para que isto seja possível, foi necessário implementar uma série de algoritmos que possibilitam a realização de uma predição do estudante e a partir disto gerar recomendações assertivas, aumentando as chances de aprovação, tendo em vista que o mesmo contará com uma gama de OA que permitirão um melhor desempenho dentro de um semestre letivo.

O funcionamento do *framework*, exibido na Figura 1, pressupõe a existência de um repositório de dados com informações básicas do aluno, da etapa de ensino que o aluno está cursando e do plano de ensino da disciplina para utilização na predição e geração de recomendações. Além das informações básicas relacionadas aos alunos, é necessário determinar o perfil de aprendizagem de cada um destes estudantes, tendo em

vista que o processo de recomendação analisará estas informações para direcionar os OA mais assertivos. A definição do perfil de aprendizagem, a partir dos perfis definidos no modelo de Felder e Silverman, é de responsabilidade da aplicação que utiliza o *framework*. A seguir cada etapa realizada no *framework* é detalhada.

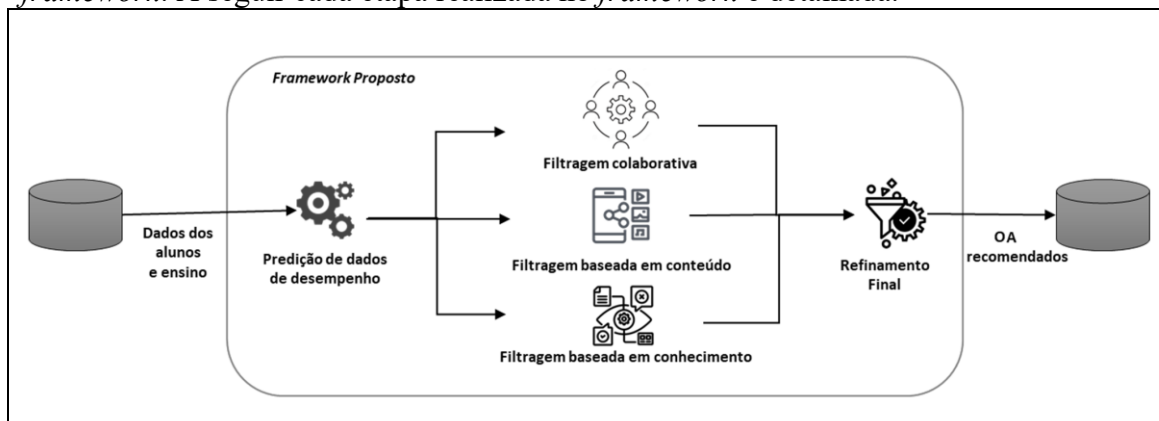


Figura 1. *Framework Proposto*

Predição de desempenho

A predição de desempenho é realizada por meio de um método que implementa três algoritmos de aprendizagem de máquina (K-Nearest Neighbors [Aha *et al.* 1991], Random Forest [Diniz *et al.* 2013] e Naive Bayes [Xu 2016]) com dados de treinamento. Estes dados de treinamento contêm informações básicas do estudante, como: *id* e nome, o perfil de aprendizagem deste aluno (conforme os atributos de Felder e Silvermann), o desempenho histórico de toda a base de dados persistida e o desempenho atual do estudante na disciplina. Com estes dados, os algoritmos são treinados para que seja possível prever o desempenho futuro do estudante. Após o treinamento do algoritmo, são lidos dados atuais do estudante e então definido o risco de reprovação do mesmo.

Ao final da execução de cada algoritmo de predição de dados, é gerado o cálculo de Acurácia, Precisão, Recall e F1, com apoio das bibliotecas *accuracy_score*, *precision_score*, *recall_score*, *f1_score* do pacote *sklearn.metrics*.

Para garantir que a previsão foi realizada com os melhores recursos para o cenário, o método de predição verifica qual dos algoritmos apresentou uma acurácia melhor na execução do processamento dos dados do aluno. A partir dessa verificação, é definida a origem dos dados previstos para o estudante, como a média prevista e a definição se o aluno será aprovado ou não.

Os dados gerados pelo melhor algoritmo de predição são submetidos a um cálculo de margem de erro considerando o percentual de acurácia do melhor algoritmo, no qual considera-se que o aluno pode ter um risco de reprovação ainda maior caso a margem de acerto do algoritmo seja mais distante de 100%. Ao final deste processamento, os dados são persistidos em uma entidade do banco de dados para registro da média calculada, juntamente com a definição de aprovação ou reprovação por parte do algoritmo preditivo.

Com os dados obtidos da predição, é possível alimentar as filtragens e realizar um processo de recomendação ainda mais refinado, aumentando a quantidade de materiais, reduzindo nível de complexidade, buscando outros tipos de materiais, com o objetivo de reforçar a aprendizagem do estudante, tendo em vista que essa previsão

pode acabar evidenciando um possível risco de reprovação que ainda não é conhecido no momento.

Filtragem Colaborativa

Nesta etapa o *framework* identifica estudantes com perfil de aprendizagem em comum por meio do cálculo da distância euclidiana. Com a lista de estudantes identificada, o algoritmo realiza um levantamento dos OA utilizados e avaliados por estes estudantes. Nesse momento, existem duas listas separadas, contendo alunos e contendo OA e a avaliação de cada estudante. Então, o algoritmo faz o cruzamento dos dados e gera uma matriz. Com a matriz de dados realizada, é aplicado o algoritmo KNN para identificar o vizinho mais próximo pela distância Euclidiana. Nesse momento, o algoritmo verifica quais os potenciais OA que podem ser recomendados ao estudante com base nas avaliações dos estudantes com perfil semelhante.

Filtragem Baseada em Conteúdo

Os algoritmos de filtragem baseada em conteúdo fazem a busca a partir do perfil de aprendizagem do estudante e dos OA que são relacionados ao plano de ensino da disciplina. Esse modelo de filtragem inicial faz com que o problema de que não existam dados suficientes para realizar a filtragem colaborativa seja superado, pois esta etapa faz a busca dos objetos de aprendizagem que de alguma forma relacionam-se ao que foi detalhado no plano de ensino do estudante, com uso de uma filtragem que analisa o texto do plano de ensino disponibilizado para os estudantes. Nessa etapa são considerados: nível de complexidade do objeto de aprendizagem, título do objeto de aprendizagem, descrição do objeto de aprendizagem e palavras-chaves do objeto de aprendizagem. Em todas as filtrações, são considerados apenas os OA que estão dentro da complexidade sugerida ao aluno, além do fato de que os objetos devem ser relacionados a etapa de ensino atual do estudante.

Filtragem Baseada em Conhecimento

Até este momento, o *framework* contém uma lista de OA que podem ser recomendados para cada estudante. Se existirem dados de desempenho do estudante na base de conhecimento, essa etapa de filtragem realiza a filtragem dos materiais identificados até o momento com base no que o estudante sabe sobre certo assunto, ou seja, nesse momento o algoritmo faz a filtragem apenas para buscar os materiais cuja complexidade está de acordo com o que o estudante conhece do assunto.

Refinamento Final

Até este momento, foram geradas recomendações pelos métodos de filtragem colaborativa, conteúdo e conhecimento. Estas recomendações estão mantidas em uma entidade de recomendações temporárias, pois necessitam de um refinamento para então gerar as recomendações oficiais. Para gerar as recomendações finais são aplicados cálculos para definição do *score* de cada recomendação, o que determina a relevância desta recomendação para o aluno. O cálculo da relevância da recomendação leva em consideração quantos métodos de filtragem recomendaram o OA (*score* é incrementado a cada seleção), nível de complexidade do OA, e se esta já foi realizada no passado.

Com as recomendações oficiais inseridas na base de dados, o aluno possuirá os dados necessários para uso dos objetos de aprendizagem recomendados. Cabe a

aplicação que está integrada ao *framework* apresentar estas recomendações para uso por parte dos alunos e registrar no banco de dados a avaliação do OA realizada pelo aluno.

O processo de recomendação pode ser executado diversas vezes para o mesmo aluno, pois o estudante avançará as etapas conforme o plano de ensino, apresentará novas interações e desempenho conhecido, o que acaba gerando novas variáveis para realizar recomendações atualizadas para o aluno.

O *framework* proposto que foi implementado por meio da linguagem de programação Python. Para Carvalho (2023), a linguagem de programação Python tornou-se uma das linguagens de programação mais populares do mundo nos últimos anos. Em relatório publicado por *Python Developers Survey 2021 Results*, é possível evidenciar que a área mais utilizada com Python é a de análise de dados, desta forma, considerando a sua grande relevância na área de análise de dados, o *framework* proposto foi implementado nesta linguagem de programação, aproveitando os recursos e otimizações disponíveis. O banco de dados utilizado foi o PostgreSQL, nele são armazenados os dados necessários para a geração das recomendações, bem como as recomendações realizadas e suas avaliações.

5. Validação

Para validação do *framework*, foram utilizados cenários para simular diferentes situações para que sejam geradas as recomendações. Com a aplicação de cenários, é possível identificar as recomendações geradas e então comparar se realmente estes dados são relevantes para o cenário aplicado. Dessa forma, é possível garantir o funcionamento do *framework* antes mesmo de aplicá-lo em um cenário real, como uma turma de alguma universidade e escola. Além disso, neste modelo de validação, é possível manter um ambiente controlado, tendo ciência da entrada de dados, simular o processamento e então comparar se a saída está de acordo com o planejado, o que seria mais complexo em um ambiente real.

A primeira etapa consiste em verificar se as filtragens estão buscando os OA apropriados ao contexto onde o aluno está aplicado. Para esta validação foram elaborados cenários de testes. Estes cenários foram desenhados pensando nos perfis de aprendizagem apresentados por Felder e Silverman. A partir destas informações, foram gerados dados de alunos considerando os modelos de aprendizagem. Já os OA foram gerados respeitando o padrão LOM, que busca identificar os objetos de aprendizagem por meio do uso de metadados. Os metadados que descrevem os objetos são definidos por propriedade e valor. Cada OA contém um conjunto de propriedades relacionadas, como: título, data de criação, autor (IEEE/ITSC, 2002). Com ambas as entidades especificadas (padrões de aprendizagem e objetos de aprendizagem), foi utilizado um padrão de plano de ensino para permitir o cruzamento entre os OA e o que será abordado ao estudante no decorrer da disciplina.

A geração dos dados fictícios de alunos (nome, desempenho, preferências, perfil de aprendizagem), foram gerados com o apoio de duas APIs disponíveis na *Web*, o chat GPT e uma segunda API (<https://api.invertexto.com/>) responsável por gerar mais dados para análise e recomendações, o que possibilitou a existência de dados suficientes para validação de cada elemento.

O método de geração de objetos de aprendizagem, juntamente com os metadados que definem os objetos. foi feito via integração com o chat GPT, para isso é

realizada uma requisição para a API do chat solicitando uma lista de objetos sobre um tema específico e então persistido na base de dados. O processo de validação do algoritmo foi realizado a partir de um plano de ensino da disciplina de Engenharia de Software, logo todos os OA gerados possuem vínculo com temas relacionados a esta disciplina. Contendo o OA cadastrado na base de dados, é necessária a geração dos metadados para identificação destes objetos de aprendizagem. OA são lidos pelo *framework* desde que existam os detalhamentos necessários conforme proposto pelo padrão LOM. Este padrão determina que os objetos de aprendizagem são descritos por meio de metadados. Neste cenário, foi desenvolvido um método exclusivo para geração dos metadados, também integrado ao chat GPT, a fim de detalhar ao máximo os objetos de aprendizagem criados. Ao todo, o método insere cerca de 20 metadados para cada objeto de aprendizagem cadastrado.

Um dos dados mais importante para o correto funcionamento do *framework* são os dados históricos. Estes dados históricos são utilizados pelos algoritmos de predição para analisar e determinar o risco de reprovação de alunos que estão cursando uma disciplina. Sem os dados históricos, os algoritmos de predição não terão capacidade de informar se o aluno possui algum tipo de chance de reprovação prévia. Para que se tenham os dados necessários, foi implementado um método que gera cinco notas de estudantes, juntamente com a informação se estes foram aprovados no passado ou não.

Para que seja possível realizar a filtragem colaborativa, foi necessário implementar um método para que sejam geradas avaliações por alunos para objetos de aprendizagem, dessa forma, haverá dados suficientes para identificar objetos de aprendizagem que se encaixam nas necessidades do aluno que será utilizado para realizar recomendações por parte do *framework*.

Com os dados simulados, foi realizada a validação unitária de cada componente:

- Filtragem baseada em conteúdo: verificação se os OA correspondem ao que o estudante precisa aprender;
- Filtragem colaborativa: verificação se o algoritmo identificou alunos com perfil de aprendizagem semelhante e buscou os OA que foram aprovados por estes alunos;
- Filtragem baseada em conhecimento: verificação se o algoritmo identificou o desempenho e cruzou com os OA disponíveis na base de dados, levando em consideração o que o aluno conhece acerca do assunto;
- Predição: verificação se o módulo preditivo foi capaz de gerar um *score* de risco de reprovação do estudante após o treinamento do modelo.

Tendo sucesso em cada etapa, o *framework* foi executado como um todo, combinando os métodos de filtragem e a geração de dados futuros a partir do módulo preditivo. Para geração das recomendações, foi formada uma base de dados com 10.271 alunos e 6.164 objetos de aprendizagem, cada um contendo os seus metadados para descrição dos objetos e dos alunos. A Tabela 1 apresenta a lista de cenários gerados e aplicados sob o *framework* para coleta das recomendações.

Cada etapa de ensino se refere a um momento do plano de ensino que o aluno se encontra no momento. Após a execução do *framework* para cada cenário, foram

coletadas as recomendações para que seja feita a verificação se as recomendações geradas realmente atendem aquilo que estava proposto pelo estudo.

Tabela 1. Definição dos Cenários de Teste

Seq	Perfil de Aprendizagem	Etapa	Aprov.	Nível	Acurácia	Precisão
1	Sensorial, Visual, Ativo e Sequencial	2	False	2	0.94	0.99
2	Sensorial, Visual, Ativo e Sequencial	2	False	1	0.94	0.99
3	Intuitivo, Visual, Ativo e Sequencial	3	True	2	0.93	0.98
4	Intuitivo, Verbal, Ativo e Sequencial	4	True	2	0.93	0.99
5	Intuitivo, Verbal, Reflexivo e Sequencial	5	True	3	0.93	0.98
6	Intuitivo, Visual, Reflexivo e Global	2	False	1	0.93	0.95
7	Sensorial, Visual, Reflexivo e Global	3	True	2	0.93	0.98
8	Intuitivo, Visual, Ativo e Global	4	False	1	0.93	0.98
9	Sensorial, Visual, Ativo e Global	4	True	2	0.94	0.98
10	Intuitivo, Verbal, Reflexivo e Global	Final	True	2	0	0

Com base nos testes em cada um dos cenários, foi feito um cálculo para verificar quantos critérios foram totalmente atendidos nas etapas de filtragem. A Figura 2 apresenta a pontuação obtida em cada um dos cenários aplicados na validação. Pode-se evidenciar que a maior parte dos cenários superou os 95% de acerto em relação ao teor das recomendações, nível de complexidade dos OA, perfil de aprendizagem relacionado aos objetos de aprendizagem, relevância para o *framework* e a quantidade total de recomendações geradas.

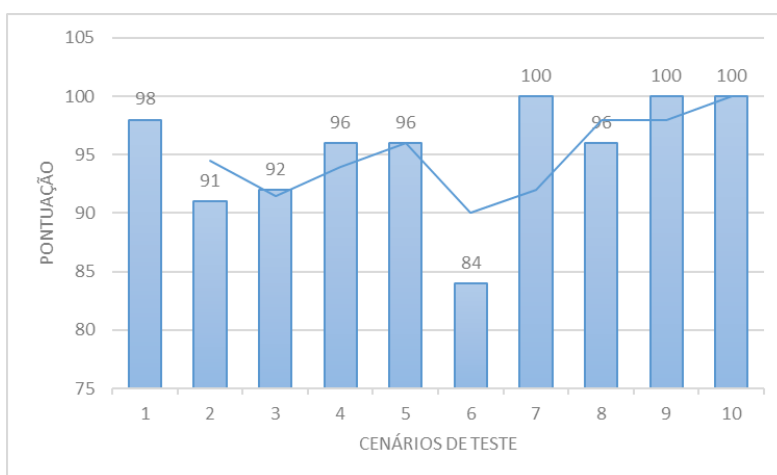


Figura 2. Gráficos de Pontuação de Cenários

A Figura 3 apresenta as combinações de recomendações possíveis e as recomendações oficialmente geradas após a execução de todos os algoritmos do *framework*. É possível identificar que em cada um dos cenários, uma quantidade considerável de objetos de aprendizagem foi, em algum momento da execução dos algoritmos, contabilizada como possíveis recomendações para o aluno, entretanto, ao final do processo, apenas a quantidade destacada em laranja realmente foi recomendada para o aluno. Por meio deste gráfico, é possível evidenciar o papel do *framework* em

busca dos objetos de aprendizagem que mais se encaixam nas necessidades do aluno, seguindo todas as combinações e parametrizações aplicadas para geração das recomendações. Não foram geradas recomendações para o cenário de teste 10 porque o aluno já finalizou a disciplina.

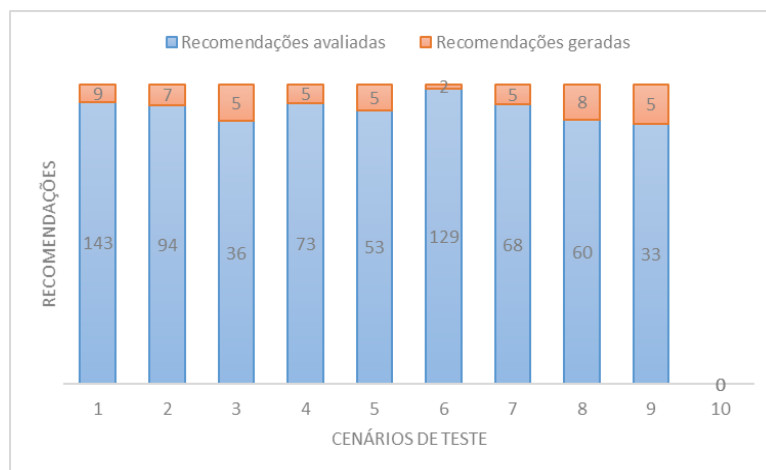


Figura 3. Gráficos de Recomendações

6. Conclusão

Na literatura foram encontrados trabalhos em que as recomendações são geradas após o conhecimento dos padrões de cada estudante, porém esta pesquisa apresenta uma alternativa para antecipar eventuais dificuldades no aprendizado de cada aluno. Os algoritmos implementados fazem com que seja possível trabalhar com dados passados para projetar o futuro e então realizar recomendações que poderão ser importantes no processo de aprendizagem, não se limitando a um nível específico de ensino, sendo possível implementar em vários cenários que permitem o uso de recursos digitais.

Como contribuição, o estudo apresenta o conceito de predição que pode ser incrementado ao processo de recomendação de objetos de aprendizagem. Os algoritmos de filtragem podem ser acoplados ao processo de predição para garantir que as recomendações geradas realmente apresentem um impacto positivo no processo de aprendizagem do estudante.

Como limitação da pesquisa, fica a ausência de uma validação em um cenário real para aplicação de um estudo ou análise que possam comprovar o benefício ao receber as recomendações geradas pelo *framework* proposto. Além desta limitação, o *framework* proposto necessita de dados para que as recomendações sejam geradas, como alunos, objetos de aprendizagem e dados históricos. Estes dados devem existir previamente em uma base de dados para uso por parte dos algoritmos de recomendação, não sendo de responsabilidade do *framework* a geração destes dados na base de dados, caso contrário, o funcionamento dos algoritmos de filtragem e recomendação serão impactados, não sendo aplicados em sua totalidade. Portanto, sugere-se como trabalho futuro formar uma base de conhecimento e então gerar as recomendações para alunos de uma instituição de ensino. Com essa aplicação, pode ser realizado um estudo para comparar o desempenho de estudantes que tiveram auxílio das recomendações e alunos que não fizeram parte do estudo.

Referências

- Aguiar, J., Fechine, J., & Costa, E. (2019, November). Experimentando a influência dos traços de personalidade do modelo big five na recomendação de recursos educacionais. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)* (Vol. 30, No. 1, p. 1711).
- Aha, D. W., Kibler, D., & Albert, M. K. (1991). Instance-based learning algorithms. *Machine learning*, 6, 37-66.
- Júlia Marques Carvalho da Silva, André Luís Alice Raabe. (2021). Um ambiente para atendimento as dificuldades de aprendizagem de algoritmos. XXV Congresso da Sociedade Brasileira de Computação, UNISINOS, São Leopoldo.
- Balabanović, M., & Shoham, Y. (1997). Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40(3), 66-72.
- Barreto, A. E., & Pozzatti, F. M. L. (2014). Objetos de aprendizagem teoria e prática. *Objetos de aprendizagem: conceitos básicos*. Evangraf, Porto Alegre.
- Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-based systems*, 46, 109-132.
- Campos, A. D., Hollerweger, L., Santos, G., Farias, A. F., & Behar, P. A. (2017). Mapeamento de soluções tecnológicas em sistemas de recomendação educacionais em âmbito brasileiro. *Informática na educação: teoria & prática*. Porto Alegre. Vol. 20, n. 3 (set./dez. 2017), p. 78-93.
- Carvalho, Carolina. (2023). O que é python? História, Sintaxe e um Guia para iniciar na Linguagem. Disponível em: https://www.alura.com.br/artigos/python?gclid=Cj0KCQjw4NujBhC5ARIsAF4Iv6davw2Ulo-B-3j_efgV6TR8smR3UOhOV6HCGd0wvaXnBdtHYM7-VfEaAppjEALw_wcB. Acesso em 31 mai. 2023.
- Carvalho, Vitor; Dorça, Fabiano. (2014). Uma Abordagem para Recomendação Automática e Dinâmica de Objetos de Aprendizagem Baseada em Estilos de Aprendizagem e em Metadados no padrão IEEE LOM. In: *Workshop de Desafios da Computação Aplicada à Educação (Desafie!)*, 3., 2014, Brasília. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, p. 47-56.
- Cazella, S., Behar, P., & Schneider, D. (2012). Ketia da Silva, and Rodrigo Freitas. Desenvolvendo um Sistema de Recomendação de Objetos de Aprendizagem baseado em Competências para a Educação: relato de experiências. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)* (Vol. 23, No. 1, pp. 26-30).
- Diniz, F. A., Neto, F. M. M., Júnior, F. D. C. L., & Fontes, L. M. O. (2013). RedFace: um sistema de reconhecimento facial baseado em técnicas de análise de componentes principais e autofaces. *Revista Brasileira de Computação Aplicada*, 5(1), 42-54.
- Drachsler, H., Verbert, K., Santos, O. C., & Manouselis, N. (2015). Panorama of recommender systems to support learning. *Recommender systems handbook*, 421-451.

- Felder, R. M.; Henriques, E. R. (1995) Learning and Teaching Styles In: Foreign and Second Language Education. *Foreign Language Annals*, v. 28, n. 1, p. 21–31.
- Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12), 61-70.
- Guttman, R. H., Moukas, A. G., & Maes, P. (1998). Agent-mediated electronic commerce: A survey. *The Knowledge Engineering Review*, 13(2), 147-159.
- Hoic-Bozic, N., Dlab, M. H., & Mornar, V. (2015). Recommender system and web 2.0 tools to enhance a blended learning model. *IEEE Transactions on education*, 59(1), 39-44.
- Imran, H., Belghis-Zadeh, M., Chang, T. W., Kinshuk, & Graf, S. (2016). PLORS: a personalized learning object recommender system. *Vietnam Journal of Computer Science*, 3, 3-13.
- Jung, C. (1971). Tipos Psicológicos. Tradução Álvaro Cabral. Rio de Janeiro. *Vozes*.
- Kolb, D. A. (2014). *Experiential learning: Experience as the source of learning and development*. FT press.
- Learning Technology Standards Committee Learning Object Metadata (IEEE LTSC LOM). (2002). IEEE Learning Technology Standards Committee (LTSC): standard for Learning Object Metadata (LOM).
- Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. *Recommender systems handbook*, 73-105.
- Maria, S. A. A., Cazella, S. C., & Behar, P. A. (2019). Sistemas de Recomendação: conceitos e técnicas de aplicação. *Recomendação Pedagógica em Educação a Distância*, 19-47.
- Marques, Gerson Adriano. (2022). Um modelo de sistema de recomendação de atividades complementares para capacitação profissional do aluno de graduação. UNISINOS, São Leopoldo.
- Moraes, Thayron Crystian Hortences, and Itana Stiubiener. (2018). Uma Abordagem Híbrida Baseada no Estilo de Aprendizagem para Recomendação de Objetos de Aprendizagem.
- Pinho, P. C. R., Barwaldt, R., Espíndola, D., Torres, M., Pias, M., Topin, L., ... & Oliveira, M. (2019, October). Developments in Educational Recommendation Systems: a systematic review. In *2019 IEEE Frontiers in Education Conference (FIE)* (pp. 1-7). IEEE.
- Santos, T. G. L. Dos. (2020). Sistemas de recomendação para o AVA moodle: uma abordagem baseada em filtragem colaborativa e na taxonomia revisada de bloom. Disponível em: <<https://repositorio.ufpb.br/jspui/handle/123456789/18269>>. Acesso em: 3 jul. 2023.
- Sweeney, M., Rangwala, H., Lester, J., & Johri, A. (2016). Next-term student performance prediction: A recommender systems approach. *arXiv preprint arXiv:1604.01840*.

- Thorat, P. B., Goudar, R. M., & Barve, S. (2015). Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, 110(4), 31-36.
- Xu, S. (2018). Bayesian Naïve Bayes classifiers to text classification. *Journal of Information Science*, 44(1), 48-59.