

Sistemas de recomendação para o AVA Moodle: uma abordagem baseada em filtragem colaborativa e na Taxonomia Revisada de Bloom

Tarsio Gomides Lima dos Santos



CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

João Pessoa, 2020

Tarsio Gomides Lima dos Santos

Sistemas de recomendação para o AVA Moodle: uma abordagem baseada em filtragem colaborativa e na Taxonomia Revisada de Bloom

Dissertação apresentada à Coordenação do Curso de Pós-Graduação em Informática da Universidade Federal da Paraíba, como requisito para a obtenção do grau de Mestre em Informática

Orientador: Prof. Dr. Claurton Siebra
Co-orientadora: Profa. Dra. Danielle R.D. Silva

Janeiro de 2020

Catálogo na publicação
Seção de Catalogação e Classificação

S237s Santos, Tarsio Gomides Lima Dos.

Sistemas de recomendação para o AVA Moodle: uma abordagem baseada em filtragem colaborativa e na Taxonomia Revisada de Bloom / Tarsio Gomides Lima Dos Santos. - João Pessoa, 2020.
78 f.

Orientação: Claurirton de Albuquerque Siebra.
Coorientação: Danielle Rousy Dias da Silva.
Dissertação (Mestrado) - UFPB/de Informática.

1. Educação Adaptativa. 2. Filtragem Colaborativa. 3. Sistemas de Aprendizagem Adaptativos. 4. Sistemas de Recomendação. 5. Taxonomia Revisada de Bloom. I. Siebra, Claurirton de Albuquerque. II. Silva, Danielle Rousy Dias da. III. Título.

UFPB/BC



UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA



Ata da Sessão Pública de Defesa de Dissertação de Mestrado de TÁRSIO GOMIDES LIMA DOS SANTOS, candidato ao título de Mestre em Informática na Área de Sistemas de Computação, realizada em 31 de janeiro de 2020.

1 Aos trinta e um dias do mês de janeiro, do ano de dois mil e vinte, às quatorze horas, no
2 Centro de Informática da Universidade Federal da Paraíba, em Mangabeira, reuniram-se os
3 membros da Banca Examinadora constituída para julgar o Trabalho Final do Sr. Társo
4 Gomides Lima dos Santos, vinculado a esta Universidade sob a matrícula nº 20181001033,
5 candidato ao grau de Mestre em Informática, na área de "Sistemas de Computação", na linha
6 de pesquisa "Computação Distribuída", do Programa de Pós-Graduação em Informática, da
7 Universidade Federal da Paraíba. A comissão examinadora foi composta pelos professores:
8 Claurton de Albuquerque Siebra (PPGI-UFPA), Orientador e Presidente da Banca, Natasha
9 Correia Queiroz Lino (PPGI-UFPA), Examinadora Interna, Danielle Rousy Dias da Silva
10 (UFPA), Examinadora Externa ao Programa, Marília Aranha Freire (IFRN), Examinadora
11 Externa à Instituição. Dando início aos trabalhos, o Presidente da Banca cumprimentou os
12 presentes, comunicou aos mesmos a finalidade da reunião e passou a palavra ao candidato
13 para que o mesmo fizesse a exposição oral do trabalho de dissertação intitulado: "Sistemas
14 de recomendação para o AVA Moodle: uma abordagem baseada em Filtragem colaborativa
15 e na Taxonomia Revisada de Bloom". Concluída a exposição, o candidato foi arguido pela
16 Banca Examinadora que emitiu o seguinte parecer: "aprovado". Do ocorrido, eu, Ruy Alberto
17 Pisani Altafim, Coordenador do Programa de Pós-Graduação em Informática, lavrei a
18 presente ata que vai assinada por mim e pelos membros da banca examinadora. João
19 Pessoa, 31 de janeiro de 2020.

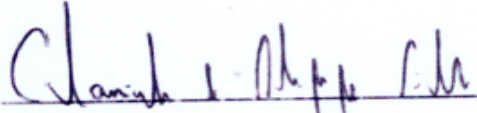
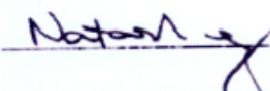
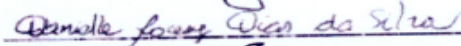
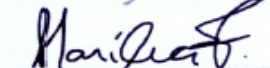

Prof. Dr. Ruy Alberto Pisani Altafim

Prof. Claurton de Albuquerque Siebra
Orientador (PPGI-UFPA)

Profª. Natasha Correia Queiroz Lino
Examinadora Interna (PPGI-UFPA)

Profª. Danielle Rousy Dias da Silva
Examinadora Externa ao Programa (UFPA)

Prof. Marília Aranha Freire
Examinadora Externa à Instituição (IFRN)



CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

Dissertação para o Programa Pós-Graduação em Informática intitulado ***Sistemas de recomendação para o AVA Moodle: uma abordagem baseada em filtragem colaborativa e na Taxonomia Revisada de Bloom*** de autoria de Tarsio Gomides Lima dos Santos, aprovado pela banca examinadora constituída pelos seguintes professores:

Danielle Rousy

Universidade Federal da Paraíba - UFPB

Natasha Queiroz

Universidade Federal da Paraíba - UFPB

Marília Freire

Instituto Federal Rio Grande do Norte - IFRN

Prof. Dr. Ruy Altafim

Coordenador do Programa Pós-Graduação em Informática - PPGI

João Pessoa, 03 de Janeiro de 2020

Centro de Informática, Universidade Federal da Paraíba
Rua dos Escoteiros, Mangabeira VII, João Pessoa, Paraíba, Brasil CEP: 58058-600
Fone: +55 (83) 3216 7093 / Fax: +55 (83) 3216 7117

Tudo posso naquele que me fortalece (Filipenses 4:13)

DEDICATÓRIA

Dedico este trabalho a minha mãe Cássiae e ao meu pai José que durante todo o processo estiveram comigo e me deram apoio para seguir na caminhada mesmo perante qualquer dificuldade encontrada.

AGRADECIMENTOS

Primeiramente agradecer a Deus que é o principal. A professora Danielle e professor Claurton pelos ensinamentos, apoio, dedicação, paciência para me ajudar a corrigir meus erros e por tudo que fizeram para me ajudar. A ex-aluna de mestrado Janalívia com quem tive a oportunidade de trabalhar em alguns momentos, me proporcionando grande aprendizado.

Por fim, agradeço a todos que participaram dessa jornada de forma direta ou indireta.

RESUMO

No cenário atual, a educação a distância e até mesmo a presencial vem fazendo uso dos ambientes virtuais de aprendizagem como ferramenta. Porém, geralmente essas ferramentas não se adaptam de acordo com o aluno, como é o caso do Moodle. Com a finalidade de resolver o problema para as chamadas questões avaliação do Moodle, esse trabalho propõe e desenvolve uma API REST para gerar recomendações usando a Taxonomia Revisada de Bloom, filtragem colaborativa baseada no usuário e também um módulo de eficiência. O recomendador usou dados de um curso de LC-EAD da UFPB e os resultados obtidos foram as recomendações para um usuário alvo. Logo, a utilização deste serviço pode trazer benefícios para estudantes, que teriam atividades sugeridas de acordo com seu perfil e nível cognitivo, e provavelmente ajudaria no aprendizado e em consequência ter uma redução da evazão.

Palavras-chave: Educação Adaptativa. Filtragem Colaborativa. Sistemas de Aprendizagem Adaptativos. Sistemas de Recomendação. Taxonomia de Bloom Revisada

ABSTRACT

In the current scenario, distance and even face-to-face education has been using virtual learning environments as a tool. However, generally these tools do not adapt according to the student, as is the case with Moodle. In order to solve the problem for the so-called Moodle evaluation questions, this work proposes and develops a REST API to generate recommendations using Bloom's Revised Taxonomy, collaborative user-based filtering and also an efficiency module. The recommender used data from an LC-EAD course at UFPB and the results obtained were recommendations for a target user. Therefore, a use of this service can bring benefits to students, who use suggested activities according to their profile and cognitive level, and can help in the learning and and consequently have a reduction in dropout.

Key-words: Adaptive Education. Adaptive Learning Systems. Collaborative Filtering. Recommender Systems. Revised Bloom Taxonomy

LISTA DE FIGURAS

1	Classificação dos Sistemas de Filtragem da Informação	25
2	Formação da Vizinhança	29
3	Representação do Funcionamento do Coeficiente de Tanimoto	32
4	Etapas básicas a serem seguidas pelos SRs	33
5	Matriz Bidimensional	36
6	Matriz Bidimensional	36
7	Detalhamento da representação do nome do Moodle	42
8	Estatísticas gerais	43
9	Países com maior número de registros	43
10	Modelo Proposto - Fonte: [Carneiro 2018]	46
11	Arquitetura do Analisador - Fonte: [Carneiro 2018]	47
12	Propriedades que Representam os Estados das Questões - Fonte: Moodle .	48
13	Tabela da Revisão Sistemática em SR	50
14	Plugin de bloco Recommender - Fonte: [Moodle 2013]	51
15	Arquitetura do gerador de recomendações híbridas	51
16	Equação para cálculo dos graus de utilidade híbrida	52
17	Plugin de bloco Analytics and Recommendations em funcionamento	53
18	Arquitetura geral do recomendador	57
19	Arquitetura do recomendador após a integração	61
20	Distribuição de questões RBT por disciplina	64
21	Distribuição de questões RBT por disciplina usadas nas avaliações	65
22	Contagem de questões RBT por disciplina que foram acertadas nas avaliações	65
23	Entrada do módulo de filtragem colaborativa baseado no usuário	66
24	Similaridade do Aluno 9 em relação ao restante dos alunos	67
25	Entrada do módulo de eficiência	68
26	Entrada do módulo de filtragem colaborativa baseado no usuário	68
27	Similaridade do Aluno 9 em relação ao restante dos alunos	68
28	Resultado parcial após a aplicação da vizinhança	69

29	Recomendações geradas pela API	69
----	--	----

LISTA DE TABELAS

1	Pesos representando o interesse de um único usuário por gênero de filmes .	27
2	Avaliação dos usuários para os filmes	29
3	Ferramentas para interatividade e interação	38
4	Características de algumas plataformas de AVAs	41
5	Questões resolvidas e acertadas por alunos	60

LISTA DE ABREVIATURAS

AVA - Ambiente Virtual de Aprendizagem

EAD - Ensino a Distância

FC - Filtragem Colaborativa

IA - Inteligência Artificial

SR - Sistemas de Recomendação

RI - Recuperação da Informação

SRI - Sistemas de Recuperação da Informação

FI - Filtragem da Informação

RBT - Taxonomia de Bloom Revisada

Sumário

1	INTRODUÇÃO	19
1.1	Motivação	19
1.2	Definição do Problema	20
1.3	Solução	21
1.3.1	Objetivo geral	22
1.3.2	Objetivos específicos	23
1.4	Estrutura da dissertação	23
2	FUNDAMENTAÇÃO TEÓRICA	24
2.1	Sistemas de Recomendação	24
2.2	Recuperação da Informação x Filtragem da Informação	24
2.2.1	Filtragem Baseada em Conteúdo (do inglês <i>Content-Based</i>)	26
2.2.2	Filtragem Colaborativa Baseada em Memória e no Usuário	28
2.2.3	Filtragem Híbrida	33
2.3	Etapas básicas de funcionamento seguidas pelos SRs	33
2.4	Taxonomia de Bloom	34
2.5	Taxonomia de Bloom Revisada	35
2.5.1	Verbos RBT	37
2.6	Ambientes Virtuais de Aprendizagem	37
2.7	Moodle	42
3	TRABALHOS RELACIONADOS	45
3.1	Trabalho que emprega a Taxonomia de Bloom e RBT na estruturação e categorização dos dados	45
3.2	Revisão Sistemática em Sistemas de Recomendação	49
3.2.1	Resultados da Revisão em Sistemas de Recomendação	50
4	DEFINIÇÃO DA PROPOSTA DE MODELAGEM E IMPLEMENTAÇÃO	55
4.1	Metodologia	55
4.2	Modelo Proposto	56

4.2.1	Módulo RBT	57
4.2.2	Módulo de verificação de eficiência dos alunos por nível	59
4.2.3	Módulo de filtragem colaborativa baseada no usuário	59
4.2.4	Integração dos Módulos	60
4.3	Desenvolvimento	61
4.3.1	Tecnologias Usadas	61
4.3.2	Apache Mahout	61
5	Resultados	63
5.1	Estudo Exploratório	63
5.1.1	Resultados módulo RBT	64
5.1.2	Resultados do Recomendador	66
5.1.2.1	Cenário 1 - Threshold muito baixo	
	66	
5.1.2.2	Cenário 2 - Recomendação gerada normalmente	
	67	
5.2	Diferencial e novidades do presente trabalho	70
6	CONCLUSÃO E CONTINUAÇÃO DO TRABALHO	71
6.1	Contribuições	71
6.2	Desafios e limitações	71
6.3	Trabalhos futuros	73
	REFERÊNCIAS	73
	ANEXO A - ANEXOS E APÊNDICES 1	78

1 INTRODUÇÃO

Este capítulo contextualiza nossa pesquisa, apresentando os objetivos e etapas em que ela foi conduzida.

A área de estudo da nossa pesquisa está relacionada com Aprendizagem Adaptativa, que pode ser definida como aprendizagem que se adapta ao nível de aprendizagem do aluno. Essa área de pesquisa também pode ser definida como o desenvolvimento de vários modelos, métodos e ferramentas que suportem a adaptação e personalização do conteúdo de ensino de acordo com o perfil de diferentes estudantes, através da associação de conceitos de Psicologia, Pedagogia e Ciência da Computação [Lane e Roschelle 2014].

1.1 Motivação

De acordo com as Nações Unidas, a qualidade da educação é um dos principais problemas globais, juntamente com o aumento da desigualdade, segurança, crise de refugiados (mais de 65 milhões), qualidade da saúde, mudanças climáticas, privacidade de dados e outros [Dougiamas 2017].

Porém, é fato que a Web tem se tornado cada vez mais um dos principais meios de apoio no processo de ensino e aprendizagem. Um dos aspectos que reforça esta afirmação é a utilização dos Ambientes Virtuais de Aprendizagem (AVA). Esses ambientes são explicitamente projetados para apoiar o ensino a distância (EAD) disponibilizando diversos recursos para esse fim, como por exemplo, a disponibilidade de chat integrado, fóruns, grupos de trabalho, questionários online, documentos, vídeos, dentre outros recursos e tudo disponibilizado na web [COSTA et al., 2013]. Outra razão foi a institucionalização no País da modalidade EAD por meio do Decreto N° 9.057¹ de 25 de maio de 2017. Como resultado, a EAD tem sido muito usada por várias instituições em diversas áreas como meio alternativo ou complementar ao ensino presencial e tendo os AVAs e a web como suporte para proporcionar esse ensino.

Paralelamente, vemos as informações disponibilizadas diariamente na web aumentarem de forma exponencial. “De acordo com a Internet World Stats dos anos 2000 até 2009 a taxa de crescimento foi de 380% e o número de domínios alcançou segundo a internet Systems Consortium a quantia de 730 milhões em janeiro de 2010” [PRIMO; VICARI; SILVA, 2010, p. 1]. Essa grande quantidade de informações não necessariamente está relacionada com a qualidade. É possível observar materiais de péssima a excelente qualidade, e muitas vezes, para o usuário comum, é difícil julgar e encontrar materiais de qualidade para o apoio a aprendizagem.

¹Disponível em <http://www.planalto.gov.br/ccivil_03/_Ato2015-2018/2017/Decreto/D9057.htm#art24> Acessado em 10 de julho de 2017

Nos AVAs usados para a EAD também é comum encontrarmos uma grande quantidade de informações, que podem vir através de artigos, jogos educacionais, livros e outros elementos existentes no AVA. Sabendo da existência de tanta informação, então, é possível que qualquer indivíduo encontre dificuldade na realização de suas atividades através dessas ferramentas, principalmente se houver pouca ou nenhuma experiência de uso. Esse cenário se forma pelo fato das pessoas interagirem com a tecnologia e se adaptarem a ela, mas do outro lado a tecnologia não faz o mesmo. No entanto, isso tem mudado aos poucos, a medida que a interação e adaptação começa a partir de ambas as partes. Por exemplo, um sistema de recomendação interage com os dados gerados pelo usuário e se adapta com base nas inferências realizadas, conseguindo assim melhorar o funcionamento da tecnologia e o uso de quem tira proveito dela.

Uma forma de melhorar esses ambientes para os usuários com relação a grande quantidade de informações ou informações não personalizadas, se dá através de sistemas de recomendação. Eles combinam várias técnicas computacionais que fornecem sugestões de itens que podem ser usados por um determinado usuário. As sugestões fornecidas objetivam auxiliar seus usuários em vários processos de tomada de decisão, como por exemplo, quais itens comprar, que música ouvir ou quais notícias ler. O desenvolvimento de sistemas de recomendação é um esforço multidisciplinar que envolve diversos especialistas em diferentes áreas, como Inteligência Artificial, Interação Homem Máquina, Tecnologia da Informação, Mineração de Dados, Estatística, Interfaces de Usuário Adaptáveis, Sistemas de Suporte a Decisão, Marketing ou Comportamento do Consumidor, e outras [Ricci et al., 2015].

1.2 Definição do Problema

Na educação, alunos e professores também sofrem ao procurar pelo que desejam na web ou em plataformas privadas direcionadas para o ensino-aprendizagem, como os AVAs. Muitas vezes, essa busca exige deles um esforço grande até conseguirem algo que realmente atenda as necessidades. Usando algoritmos, técnicas e outros elementos existentes na área de Ssistemas de Recomendação (SR), é possível facilitar esse processo. Por exemplo, um SR pode ser utilizado para recomendar materiais didáticos mais adequados conforme as dificuldades já observadas e identificadas de um certo aluno em um dado conteúdo, ou mesmo indicar material complementar para que aquele conteúdo seja melhor assimilado, dentre muitas outras possibilidades.

Ciente disso, é importante tornar possível a obtenção e recomendação de dados de acordo com o perfil do aluno. Os SRs vem justamente com a finalidade de amenizar ou até resolver esse problema, possibilitando a criação de uma relação entre a informação obtida e o usuário [CAZELLA et al. 2009]. Os SRs tem sido vastamente reconhecidos e utilizados nos últimos anos como uma forma de prover, suporte personalizado aos alunos

enquanto realizam tarefas de aprendizagem em ambientes de aprendizagem baseados na Web [Drachsler et. al. 2015].

Tendo em mente que o Moodle é um AVA que possibilita a conexão e interação entre professores e alunos, e representa um dos AVAs mais utilizados no Brasil², então, o presente trabalho propõe a criação de um modelo de SR seguindo a arquitetura REST na forma de uma API. Tal modelo pode ser aplicado no ambiente virtual de aprendizagem Moodle, como também em outras aplicações. É possível perceber que com o Moodle funcionando sem um SR, a interação usuário-usuário e usuário-sistema acaba sendo um pouco desinteressante e mal utilizada, pois não há nenhuma inferência realizada nos dados gerados por essas interações, visando possibilitar uma melhora nas interações futuras. Os dados gerados a partir dessas interações podem oportunizar recomendações de natureza variadas, e acreditamos que, podem auxiliar no processo ensino-aprendizagem melhorando a experiência de uso desses ambientes como também auxiliando em problemas como evasão e retenção dos cursos a distância.

Levando em conta toda a situação citada anteriormente e sabendo que no Moodle por padrão não há um mecanismo que explicitamente esteja focado na busca pela melhora do processo cognitivo, então, temos como problema a ser resolvido nesse trabalho, a busca por tornar possível um certo nível de adaptação em relação as questões que são disponibilizadas para os alunos de um determinado curso, como uma tentativa de exercer influência no processo cognitivo e agregar um maior poder educacional a ferramenta. Na busca pela resolução deste problema, utilizou-se questões previamente existentes em um curso no Moodle, que para o caso deste trabalho são os cursos de licenciatura em computação a distância da UFPB. Além do contexto exposto, vale ressaltar que esse trabalho usa como etapa inicial, o resultado obtido no trabalho de [Carneiro 2018], que será detalhado na seção 3.1 do presente trabalho.

1.3 Solução

Sistemas de Recomendação possuem um papel bem definido, que é recomendar algo a alguém. Um SR aplicado em AVAs permite fornecer aos estudantes vários recursos, como por exemplo: recomendação de atividades, personalização do aprendizado, sugestão de itens interessantes e várias outras possibilidades. Do lado dos educadores, é possível realizar predição de performance, criação de grupos de alunos com características afins e várias outras possibilidades (Ferro, 2010).

Um SR torna possível selecionar itens personalizados com base nos interesses dos usuários e conforme o contexto no qual estão inseridos. A Taxonomia de Bloom classifica o usuário de acordo com o seu nível de cognição.

²Disponível em <<https://moodle.net/stats/>> Acessado em 19 de setembro de 2017

Dentre as técnicas de recomendações mais comuns para SRs, temos a recomendação baseada em Filtragem Colaborativa, Recomendação Baseada em Conteúdo e a Híbrida que mistura as técnicas citadas anteriormente, fortalecendo as fraquezas de cada uma, o que faz dela uma técnica mais poderosa (Ferro, 2010).

Neste trabalho, os conhecimentos relativos a SR foram empregados na área de educação, com o objetivo de identificar e sugerir aos estudantes questões provavelmente relevantes e personalizadas de acordo com sua classificação na Taxonomia de Bloom e também de acordo com as atividades que ele já realizou anteriormente. Caso o usuário não possua classificação na Taxonomia de Bloom, seja por não ter realizado nenhuma atividade ou por não ter acertado nenhuma delas, então, nesse caso as questões mais populares serão indicadas.

Para a geração das recomendações há uma variedade de possibilidades, nas quais é possível trazer algumas partes da Inteligência Artificial, como por exemplo mineração de dados. No entanto, para a solução do problema proposto nesse trabalho, é preciso entender que inicialmente, a Taxonomia Revisada de Bloom é utilizada como uma etapa anterior a geração das recomendações, como tentativa de medir a aprendizagem do aluno e classificá-lo em um dos níveis da taxonomia, para que posteriormente a parte que trata da geração de recomendações comece a atuar.

Para a elaboração desta solução, em um primeiro momento, foi realizado o estudo referente à fundamentação teórica com base em uma ampla pesquisa bibliográfica, buscando fundamentar conceitos importantes em relação a SRs, AVAs e a Taxonomia de Bloom, bem como um levantamento de trabalhos e sistemas já existentes nestas áreas. Após, foi proposto o modelo arquitetural do recomendador, além disso, o AVA Moodle foi o escolhido, principalmente pelo fato de ser utilizado nos cursos LC-EAD, que é exatamente o provedor dos dados necessários para a nossa proposta. Em seguida, foram selecionadas as tecnologias adotadas para a implementação da API REST, que possui o fundamental papel de gerar as recomendações, e por último partiu-se para o desenvolvimento e análise dos resultados dos experimentos realizados.

Em resumo, este trabalho solucionará o problema da recomendação de atividades aos alunos do Moodle, através do uso de um conjunto de dados educacionais, tendo como finalidade que essas recomendações possam ajudar o usuário a sempre encontrar questões que possibilitem ajudar a maximizar o seu aprendizado, sempre considerando que tais questões estão no seu nível cognitivo da taxonomia revisada de bloom.

1.3.1 Objetivo geral

Como objetivo geral do presente trabalho, pretende-se propor um modelo de SR com base em algoritmos de recomendação conhecidos na literatura, fundidos com outras

técnicas, o que por consequência torna esse recomendador híbrido, sendo ele entregue na forma de uma API REST.

1.3.2 Objetivos específicos

Como objetivos específicos pode-se elencar os seguintes pontos:

- Identificar os dados gerados no AVA Moodle que podem ser utilizados na geração de recomendações
- Sondar e identificar técnicas e algoritmos para a modelagem do SR
- Investigar os conceitos comuns inerentes a uma API REST visando entender como ela pode ser útil no presente trabalho e como ele pode ser aplicada ao recomendador
- Identificar tecnologias importantes para a construção do recomendador
- Investigar os resultados gerados pelo recomendador, de modo a saber se o comportamento está como o esperado

1.4 Estrutura da dissertação

O trabalho está estruturado em cinco seções distribuídas da seguinte forma:

- **Introdução:** Define a motivação para a produção do trabalho e fornece uma visão geral a respeito do que será abordado.
- **Fundamentação Teórica:** Explica e exemplifica os principais conceitos encontrados na literatura que são úteis para a execução do trabalho.
- **Trabalhos Relacionados:** Resume os trabalhos relacionados, que serviram como base para a contextualização deste projeto.
- **Proposta de Modelagem e Implementação:** Mostra as etapas existentes na pesquisa, modelagem e implementação, descrevendo cada uma delas.
- **Resultados:** Explica como o sistema foi feito, mostra a arquitetura usada, tecnologias envolvidas e os resultados do sistema.
- **Conclusões:** Relata as dificuldades encontradas durante a realização do trabalho, lições aprendidas e trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Sistemas de Recomendação

O estudo de SR tem como foco desenvolver modelos para reduzir a sobrecarga de informações que um usuário recebe ao procurar por algo, e isso é conseguido através de recomendações com base no perfil do usuário (ADOMAVICIUS; TUZHILIN, 2005).

SRs são amplamente usados como meio para obter elementos ainda não usados e/ou desconhecidos e que são considerados adequados para um determinado usuário, além disso, atua com objetivos diferentes dependendo da área. Por exemplo, em um e-commerce pode aumentar de maneira significativa o número de vendas, já na área da educação pode melhorar e tornar mais eficiente o aprendizado e assim por diante. Dito isto, percebe-se que um dos desafios na área de SR está em selecionar os itens para serem recomendados aos usuários em um conjunto variado e maior de itens, tendo em vista que esses podem ser em pouca quantidade ou em quantidades extremas e devem estar de acordo com as preferências dos usuários (FERRO, 2010).

Para deixar o problema mais formal e compreensível vamos supor que U seja o conjunto de usuários de um AVA qualquer e T a quantidade de atividades existentes até o momento. Como resultado temos que o produto cartesiano de $U \times T \rightarrow R$, onde R pode ser um conjunto de combinações bem extenso. Um SR também busca determinar o nível de utilidade para cada item recomendado a um usuário específico, e isso é feito calculando o grau de utilidade f , que indicará o quanto o item satisfaz o usuário. Assim, concluímos que um SR seleciona e recomenda itens que possuam maiores graus de utilidade para um determinado usuário (FERRO, 2010).

Adomavicius e Tuzhilin (2005) descrevem o processo de recomendação através da seguinte equação:

$$\forall u \in U, t'_u = \operatorname{argmax} f(u, t), t \in T \quad (1)$$

Com U representando o conjunto de usuários e T o conjunto de itens a serem recomendados, tendo também uma função de grau de utilidade f . Com isso, vemos que para cada usuário $u \in U$ um SR tem como objetivo recomendar apenas os itens $t' \in T$ que maximizem a função de grau de utilidade $f(u, t)$ de um determinado usuário.

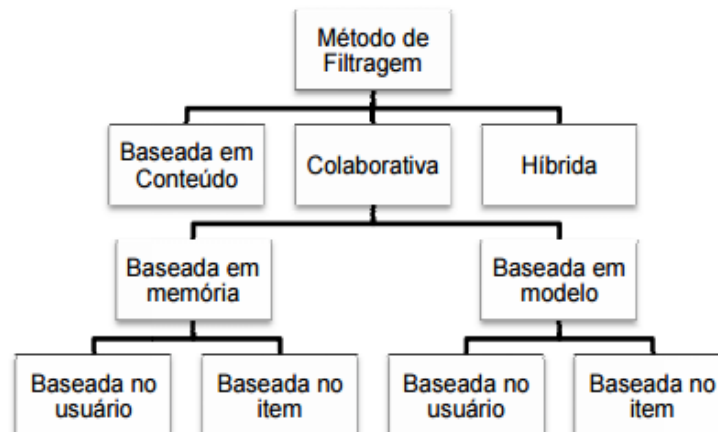
2.2 Recuperação da Informação x Filtragem da Informação

A recuperação da informação (RI) abrange estudos voltados aos processos relativos ao acesso a informação, propondo modelos que solucionem a necessidade da informação

de um usuário em determinado conjunto de dados. Os problemas da RI estão relacionados ao modelo de recuperação adotado, que geralmente desconsideram o contexto da consulta e com isso retornam documentos irrelevantes (BAEZA-YATES; RIBEIRO-NETO, 2013). Os sistemas de recuperação da informação (SRI) são compostos por três processos: a consulta da informação, a indexação e o cálculo de similaridade que verifica a ocorrência ou não dos termos da consulta na base de dados. Recentemente tem sido utilizadas técnicas de mineração de dados, mas normalmente não levam em consideração os aspectos semânticos da consulta ou do perfil do usuário.

A filtragem da informação (FI) é uma técnica utilizada para filtrar uma grande quantidade de informação e entregar para um determinado usuário apenas informações que lhe é relevante e é comumente utilizada pelos SR para conseguir fazer recomendações de itens para os usuários. Alguns métodos de FI usados em SRs são mostrados na Figura 1 (LÁZARO, 2010):

Figura 1 – Classificação dos Sistemas de Filtragem da Informação



Fonte: (LÁZARO, 2010)

Em geral, existem diferentes vertentes de classificação para SRs na literatura, a da Figura 1 é uma delas. Dos métodos existentes nas mais diversas classificações, podemos destacar três métodos de filtragem que formam um conjunto amplamente usado atualmente. Segundo [Adomavicius e Tuzhilin 2005] essa classificação é definida da seguinte maneira:

- **Sistemas de recomendação com filtragem baseada em conteúdo:** As recomendações de itens recebidas pelos usuários são realizadas com base na similaridade das características dos itens que o usuário gostou/adquiriu no passado e do item a ser indicado. Portanto, o histórico de gostos/aquisições dos itens deve ser levado em consideração para a geração de novas recomendações.
- **Sistemas de recomendação com filtragem colaborativa baseada em memória**

e no usuário: As recomendações de itens recebidas pelos usuários são realizadas com base nas preferências de outros usuários com um perfil semelhante.

- **Sistemas de recomendação híbridos:** Utiliza uma mistura das técnicas de recomendação baseada no conteúdo e recomendação baseada na filtragem colaborativa.

As três técnicas citadas, serão detalhadas em termos de funcionamento nas próximas sessões.

2.2.1 Filtragem Baseada em Conteúdo (do inglês *Content-Based*)

Nessa técnica, o histórico da interação entre usuário e sistema é analisado, tendo como objetivo obter quais elementos do sistema o usuário mais gostou ou adquiriu em seu passado. Com base nessa informação o sistema tenta inferir quais as melhores recomendações de acordo com histórico de cada usuário (FERRO, 2010). Portanto, se um usuário gosta/adquire um item X que possui certas características, conclui-se que ele irá receber recomendações de itens Y com características similares. Para exemplificar o que foi explicado, podemos pegar um situação em que um usuário tenha assistido um filme através de uma plataforma de streaming de vídeo, e o mesmo possui características como diretor, elenco, gênero, nacionalidade, data de lançamento, então, poderia ser gerada uma recomendação de um outro filme que fosse parecido com os previamente assistidos baseando-se no conjunto de gêneros.

Sistemas de recomendação baseados no conteúdo tentam prever o grau de utilidade de um determinado item para um determinado usuário, com base em seu histórico. A função de utilidade $f(u, t)$ de um produto t para um usuário u tem seu cálculo baseado na função de utilidade $f(u, t_i)$, tal que, t_i são os itens pertencentes ao conjunto T , que é o conjunto dos itens relacionados ao usuário u em momentos passados (ADOMAVICIUS; TUZHILIN, 2005). Portanto, para que uma recomendação seja feita, são comparadas as características dos itens atribuídos ao histórico com o item alvo de uma possível recomendação (FERRO, 2010).

Para que seja factível a existência de uma correlação entre o item a ser indicado e os itens do histórico do usuário, é necessário utilizar alguma ideia que permita a construção do perfil do usuário de acordo com suas interações no passado. Segundo Adomavicius e Tuzhilin (2005) um modo de se construir o perfil do usuário é armazenar um vetor de pesos, com cada peso representando uma característica do item. Na literatura existem outras maneiras alternativas ao vetor de pesos, entretanto, seu uso é muito comum nesta técnica de recomendação.

Para exemplificar essa forma de construção do perfil do usuário A , suponha que a plataforma de streaming de vídeo possua filmes com os gêneros de ação, comédia, drama,

ficção, romance e terror, e que o usuário A tenha assistido até agora filmes de ação, drama e ficção, conforme o quadro a seguir:

Tabela 1 – Pesos representando o interesse de um único usuário por gênero de filmes

Gênero	Nome	Total
Ação	filme1, filme2, filme3, filme4, filme5	5
Comédia	filme6, filme7	2
Drama	filme8	1
Ficção	<nenhum>	0

De acordo com os dados disponibilizados na tabela, esse tipo de recomendação pode usar o número total de filmes assistidos por gênero para representar o peso atribuído a cada gênero para o usuário em questão, portanto, a representação geral seria $\text{pesos}_{\text{usuario}}(\text{ação}, \text{comédia}, \text{drama}, \text{ficção})$, e no caso particular para o usuário da tabela anterior que chamaremos de X, teríamos a seguinte representação $\text{pesos}_X(5, 2, 1, 0)$. Esta abordagem, permitiria ao sistema de recomendação a identificação de uma hierarquia de qual é a preferência do usuário por gênero. Portanto, o sistema de recomendação usa o vetor de pesos para decidir quais filmes que deverão ser recomendados (FERRO, 2010).

Os sistemas que utilizam apenas esta técnica de recomendação possuem algumas desvantagens e limitações:

- **Não possibilita a recomendação de um item que possui uma boa popularidade:** Deixa o usuário preso a itens parecidos com os que ele gosta, evitando que conheça novos itens. Para o usuário X, do exemplo anterior, que em seu histórico de filmes não há nada de ficção assistido, a recomendação de um filme desse gênero que a maioria dos usuários gostam, provavelmente não seria recomendado por causa do seu gênero, pois a função de grau de utilidade $f(X, \text{filme popular de ficção})$ seria igual a zero (FERRO, 2010).
- **Possibilidades limitadas de representação do conteúdo dos itens:** Existem casos em que as informações não podem ser representadas satisfatoriamente usando apenas variáveis de escala quantitativa ou qualitativa. Com exemplo temos sons, vídeos e figuras, que são elementos com difícil extração de características (BEZERRA, 2004).
- **Análise de conteúdo limitada:** Não fornece recomendações adequadas se o conteúdo analisado não contém informações suficientes.

Segundo (SILVA, 2014), também é possível encontrar várias vantagens nessa técnica, sendo elas:

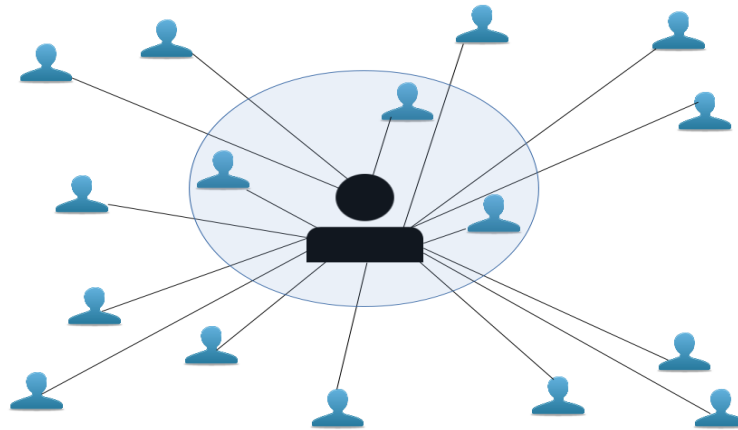
- **Necessário conhecer somente o usuário alvo da recomendação:** Não é necessário conhecê-lo em comunidade.
- **Transparência:** Transparente por não possuir dependência em relação aos dados de outros usuários para gerar a recomendação.
- **Independente das reações dos usuários:** Ao inserir um novo item no sistema, não existirá nenhuma avaliação prévia disponível, considerando as avaliações feitas pelos usuários, no entanto, por meio da verificação das propriedades do item, é possível obter a sua similaridade com outros itens, o que torna possível recomendá-lo.
- **Menor custo computacional:** Normalmente, em todos os sistemas, a quantidade de usuários é maior que a de itens, logo, identificar itens similares pode ter custo computacional menor que identificar usuários similares.

2.2.2 Filtragem Colaborativa Baseada em Memória e no Usuário

Baseia-se na análise de preferências de um conjunto de pessoas, fazendo uma correlação usuário-usuário, buscando dentre os usuários o mais similar ao usuário alvo da recomendação (CAZELLA et al., 2009). É comum na literatura a formulação do problema através de uma matriz USUÁRIOS X ITENS, com cada célula da matriz armazenando o grau de utilidade $f(u, t)$ do item t em relação ao usuário u , que é obtido pela avaliação implícita ou explícita do usuário. Nesta matriz também é comum existir células vazias para itens ainda não avaliados e o propósito é justamente prever o grau de utilidade $f(u, t)$ do item para um determinado usuário, tomando como base o grau de utilidade $f(u, t)$ do mesmo produto para um conjunto de usuários $u_i \in T$ que possuem gostos similares ao do usuário que se está tentando recomendar algo (FERRO, 2010).

Nessa técnica de recomendação, é preciso identificar os usuários com características semelhantes formando grupos de vizinhos. O processo citado anteriormente é chamado de geração de vizinhança e tem como objetivo a identificação de quais usuários são mais próximos de um usuário específico (FERRO, 2010). A figura 2 mostra os três vizinhos mais similares ao usuário representado na cor preta.

Figura 2 – Formação da Vizinhança



Fonte: Elaborada pelo autor

Segundo Herlocker (2000) e Cazella et al. (2009), para realizar recomendações através desta técnica de filtragem é comum encontrar na literatura os seguintes passos a serem seguidos:

- Calcular o nível de similaridade de todos os usuários em relação ao usuário alvo da recomendação
- Selecionar um subconjunto de usuários com maiores similaridades em relação ao usuário alvo
- Normalizar as avaliações e computar a predição dos produtos para o usuário

Considerando os três passos citados acima, construiremos a Tabela 2, que é uma tabela de USUÁRIOS X ITENS usada para exemplificar como essa técnica funciona. A tabela contém dados fictícios de um site streaming de vídeo também fictício, apenas com finalidade de auxiliar no entendimento da técnica. As células vão representar o grau de utilidade de um filme assistido e avaliado por um determinado usuário, com nota variando entre 0 e 10. Como falado anteriormente, as células vazias indicam que o usuário ainda não assistiu aquele filme, tornando ele um possível alvo de recomendação.

Tabela 2 – Avaliação dos usuários para os filmes

Usuário	Filme 1	Filme 2	Filme 3	Filme 4	Filme 5
A	3		10	1	2
B	5	3	3	2	8
C	9	9	3	9	9
D	2			4	
E	10		8	10	6

Para realizar o primeiro passo, que é o cálculo do nível de similaridade entre os usuários, é possível utilizar várias funções existentes. Algumas das funções para o cálculo de similaridade amplamente usadas em sistemas de recomendação são:

- Correlação de Pearson
- Similaridade por cosseno de vetor
- Coeficiente de Tanimoto

Agora será mostrado em mais detalhes como funciona a geração da recomendação usando a Correlação de Pearson e Coeficiente de Tanimoto, mas além das três maneiras citadas para o cálculo da similaridade, existem outras.

Segundo Herlocker (2000), para calcular a similaridade através da Correlação de Pearson, temos a seguinte equação:

$$w_{a,u} = \frac{\sum_{i=1}^m [(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)]}{\sqrt{\sum_{i=1}^m (r_{a,i} - \bar{r}_a)^2 \sum_{i=1}^m (r_{u,i} - \bar{r}_u)^2}} \quad (2)$$

Ao aplicar essa equação, será possível obter um valor que indica a correlação do usuário a em relação ao usuário u , representado por $w_{a,u}$. As variáveis $r_{a,i}$ e $r_{u,i}$ indicam a avaliação que os usuários a e u deram para o item i . Já as variáveis $\bar{r}_a$ e $\bar{r}_u$ será a média de todas as avaliações realizadas pelos usuários a e u , em relação aos itens que eles avaliaram previamente. Os resultados encontrados irão variar no intervalo de -1 até 1 (FERRO, 2010).

Após efetuar o cálculo da similaridade é preciso definir como será obtida a vizinhança, existem várias maneiras, mas aqui vamos elucidar duas delas. Uma delas é definir um threshold, por exemplo, só se deve considerar usuários com resultados de similaridade acima de um valor X . Outra maneira, seria pegar os três mais similares, ou seja, os três vizinhos mais próximos, porém, isso poderia trazer problemas em alguns casos onde os níveis de similaridade são baixos, mesmo para os vizinhos mais próximos. Para nossos exemplos, iremos usar sempre a vizinhança por threshold.

Para o caso que estamos exemplificando, usamos a vizinhança por threshold, obtemos apenas usuários que possuam um certo nível de similaridade, e isso é definido através de w_{corte} . Quanto mais próximo de 1 for o valor de w_{corte} , mais similares os usuários serão, em contrapartida, quanto mais distante de 1 for o valor de w_{corte} , menos similares serão os usuários e a quantidade de vizinhos será maior (FERRO, 2010).

A última etapa consiste em definir quais itens serão recomendados e isso é feito através do cálculo da predição de um item para um determinado usuário. Segundo FERRO (2010), o cálculo da predição pode ser feito através da equação abaixo:

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u=1}^n [(r_{u,i} - \bar{r}_u) * w_{a,u}]}{\sum_{u=1}^n w_{a,u}} \quad (3)$$

Na equação mostrada, $p_{a,i}$ é a predição do usuário a em relação ao item i , levando em consideração todo o conjunto de usuários u que são vizinhos de a . Temos ainda $\bar{r}_a$ e $\bar{r}_u$ indicando a média das avaliações dos usuários a e u e $w_{a,u}$ a similaridade. Quanto maior a quantidade de informações sobre os usuários, mais precisa será a predição (FERRO, 2010).

No caso de usar o cálculo da similaridade por cosseno, teremos a medida do cosseno do ângulo entre dois vetores. Para isso realizamos o produto escalar no denominador e no numerador a normalização dos vetores [Dezembro, 2019].

$$similarity(\vec{A}, \vec{B}) = \cos(\vec{A}, \vec{B}) = \frac{\vec{A} \bullet \vec{B}}{\|\vec{A}\| * \|\vec{B}\|} \quad (4)$$

Para o caso do cálculo da similaridade usando o Coeficiente de Tanimoto, os valores de preferência são ignorados completamente, se importante apenas se o usuário expressa uma opinião positiva ou negativa, ou seja, algo binário.

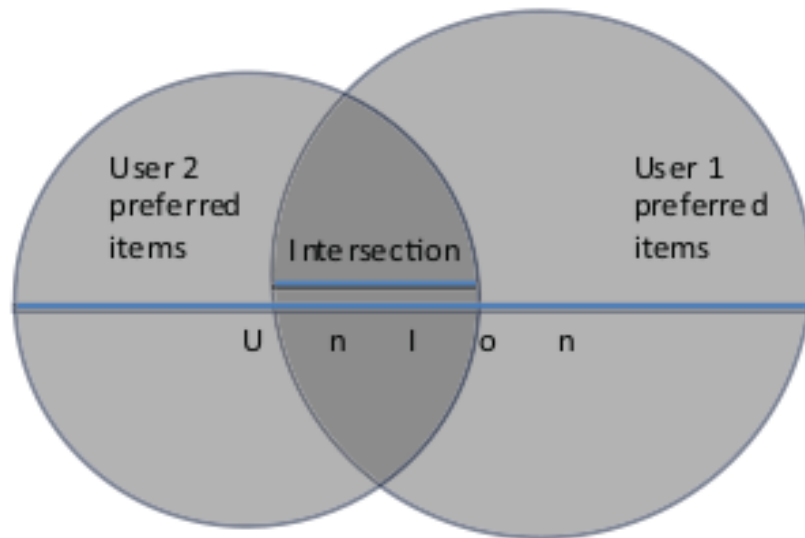
O Coeficiente de Tanimoto é a razão do tamanho da intersecção em relação ao tamanho da união dos itens que dois usuários demonstraram preferência [Owen et al., 2012]. A equação abaixo mostra como essa medida de similaridade pode ser calculada:

$$T(X, Y) = \frac{X \cap Y}{X \cup Y} \quad (5)$$

X e Y definem elementos nos conjuntos de dados e o coeficiente é definido como o tamanho da intersecção dividido pelo tamanho da união dos conjuntos de dados.

Agora representando o que foi falado anteriormente de maneira visual na figura 3, o Coeficiente de Tanimoto é o número de itens que ambos usuários expressam alguma preferência, dividido pelo número de itens que os usuários expressaram alguma preferência [Owen et al., 2012].

Figura 3 – Representação do Funcionamento do Coeficiente de Tanimoto



Após obter a similaridade, deve-se obter os itens dentro da vizinhança usando threshold ou alguma outra maneira. Com relação ao cálculo predição, não existe nesse caso, já que se trata de valores binários

Os sistemas que utilizam apenas esta técnica de recomendação, independentemente do que foi escolhido para obter a similaridade, vizinhança e predição, possuem algumas desvantagens e limitações:

- **Esparsidade:** Quando a matriz possui poucos valores atribuídos/gerados, as recomendações podem não ser boas.
- **Problema do primeiro avaliador:** Ao adicionar novos itens eles ainda não possuem avaliações dadas por nenhum usuário, conseqüentemente não podem ser recomendados até que algum usuário atribua/gere valores para esses itens.
- **Cold-Start:** Causado por novos usuários, pois eles ainda não interagiram com nenhum item no sistema, não possuem valor gerado/atribuído a nada, então não é possível calcular a sua similaridade com outros usuários.

Também é possível encontrar vantagens nessa técnica, sendo elas:

- **Produção de recomendações inesperadas:** Pode recomendar itens com características diferentes e surpreender
- **Independência do conteúdo dos itens:** Não necessita ter conhecimento sobre o conteúdo dos itens

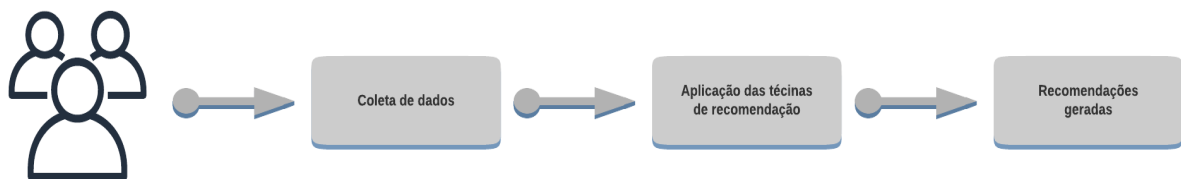
2.2.3 Filtragem Híbrida

Sistema de recomendação baseado em filtragem híbrida, realizam a combinação da recomendação com base no passado dos usuários e a recomendação com base na colaboração dos usuários, ou seja, com base nas correlação item-item e na correlação pessoa-pessoa (FERRO, 2010). Essa mistura pode proporcionar uma melhora nos problemas existentes na filtragem colaborativa e na filtragem baseada em conteúdo, onde uma corrige os problemas encontrados na outra. Porém, vale ressaltar, que outras técnicas podem ser usadas para a construção de um recomendador híbrido, não se limitando apenas as que foram citadas.

2.3 Etapas básicas de funcionamento seguidas pelos SRs

O funcionamento de um SR segue, normalmente, um conjunto de etapas básicas, como as apresentadas na figura 4.

Figura 4 – Etapas básicas a serem seguidas pelos SRs



Fonte: Elaborada pelo autor

- **Identificação do usuário:** Descobre o usuário alvo das recomendações
- **Coleta de dados:** Pode ser feita de forma **explícita** onde os usuários passam as informações geralmente através de formulários ou **implícita** onde o sistema obtém as informações automaticamente
- **Aplicação das técnicas de recomendação:** Núcleo do sistema de recomendação, que pode ser construído para utilizar na geração de suas recomendações: algoritmos de ciência cognitiva, teoria da aproximação, recuperação da informação, filtragem da informação, mineração de dados e etc.
- **Apresentação das recomendações geradas:** Exibe as recomendações para o usuário. A exibição pode ser na forma de **Push**, que não necessita que o usuário solicite a exibição das recomendações ou na forma de **Pull**, que o usuário precisa solicitar que as recomendações sejam exibidas.

Dentre os conceitos apresentados, utilizaremos a técnica de filtragem colaborativa baseada em memória e no usuário na construção do recomendador proposto nesse trabalho, mais especificamente usando a similaridade de tanimoto. O recomendador proposto será um recomendador híbrido pelo fato de misturar tal técnica de recomendação com outra construção proposta. Posteriormente será mostrao com mais detalhes essa construção.

2.4 Taxonomia de Bloom

A Taxonomia dos objetivos educacionais, conhecida também por Taxonomia de Bloom, é um framework que pode classificar declarações do que se espera ou pretende que os alunos aprendam como resultado da instrução. Esse framework foi concebido como meio de facilitar o intercâmbio de itens de avaliação entre professores de várias universidades, a fim de criar bancos de itens de avaliação, cada um medindo o mesmo objetivo educativo. A Taxonomia de Bloom apresenta um modelo que classifica os diferentes níveis de cognição humana de pensamento, aprendizagem e compreensão. Ela foi criada nos anos 50 pelo psicólogo educacional Benjamim Bloom e seus colegas. Além de facilitar o intercâmbio de questões de testes, essa taxonomia objetiva ajudar no planejamento, organização e controle dos objetivos de aprendizagem [Anderson e Krathwohl 2001].

Na Taxonomia de Bloom, os objetivos educacionais foram divididos em três domínios: cognitivo, afetivo e psicomotor. Esta pesquisa se foca apenas no domínio cognitivo da RBT, pois o estilo cognitivo tende a estar muito próximo da personalidade, se comparado com outros tipos de estilos.

O domínio cognitivo é definido como as formas de cognição da informação, como por exemplo, se o aprendiz tende a dividir a informação em partes, ou tenderá a agrupar a informação, ou ainda ir direto às conclusões a respeito da informação ou, ao contrário, ser reflexivo sobre o assunto [Souto 2003].

A classificação hierárquica dos processos cognitivos está dividida em seis níveis de complexidade, que são: (1) Avaliação, (2) Síntese, (3) Análise, (4) Aplicação, (5) Compreensão e (6) Conhecimento. Para cada nível de complexidade desta Taxonomia é exibida uma lista de verbos imperativos relacionados que são usados como suporte ao planejamento sistêmico dos cursos e ou disciplinas na estruturação dos seus objetivos, estratégias e avaliações [Costa et al. 2013].

A Taxonomia de Bloom do domínio cognitivo fornece um framework pedagógico para determinar quão difícil cognitivamente é uma questão. Usando esse framework, pode-se determinar se as habilidades cognitivas atuais de um aluno são apropriadas para a tarefa que o aluno está executando [Peckham e McCalla 2012].

2.5 Taxonomia de Bloom Revisada

No final dos anos 90, a Taxonomia de Bloom foi revisada por Anderson e Krathwohl [Anderson e Krathwohl 2001] para torná-la relevante para a aprendizagem no século XXI. A nova versão da taxonomia é conhecida como Taxonomia de Bloom Revisada, do inglês *Revised Bloom Taxonomy* (RBT). Segundo os autores, mesmo mantendo parte da estrutura original, a taxonomia revisada é mais adequada para suportar as novas formas de aprendizagem e consequentemente tirar um melhor proveito de objetivos educacionais.

Em [Jesus e Raabe 2009], as principais características da RBT são explicadas, como a seguir. A RBT consiste em duas dimensões: (1) A dimensão do conhecimento, que abrange as subcategorias da categoria “conhecimento” na taxonomia original, (2) A dimensão dos processos cognitivos que abrange as seis categorias da taxonomia original. As mudanças mais significativas do processo cognitivos foram:

- As seis categorias foram renomeadas, para melhor representar a forma como são utilizadas pelos docentes na definição de seus objetivos;
- Os aspectos verbais utilizados nas categorias Conhecimento e Compreensão foram mantidos, porém renomeados, Conhecimento tornou-se Lembrar e Compreensão foi mudado para Entender;
- A categoria Síntese tornou-se Criar. Aplicação, Análise, e Avaliação tornaram-se respectivamente Aplicar, Analisar e Avaliar;
- As categorias Síntese e Avaliação (Criar e Avaliar) foram trocadas de lugar.

A matriz bidimensional que representa a RBT pode ser visualizada na figura 5. Essa tabela deve ser utilizada com o intuito de melhor estruturar os objetivos educacionais e auxiliar os educadores na melhor elaboração do planejamento e na escolha adequada de estratégias e tecnologias educacionais. Na matriz bidimensional, a coluna vertical corresponde à dimensão conhecimento a coluna horizontal corresponde ao processo cognitivo. Os objetivos são inseridos nas células, formadas pela intersecção das dimensões. Um mesmo objetivo pode ser inserido em mais de uma célula e o preenchimento de todas as células consecutivas não é necessário, como ilustra a situação hipotética representada na tabela. Um dos desafios para se utilizar esse modelo é a dificuldade que alguns educadores encontram na utilização adequada da matriz proposta. Os verbos de ação da taxonomia original podem ser perfeitamente inseridos nas categorias correspondentes; entretanto para descrever como será alcançado esse objetivo, como também para a escolha das estratégias e tecnologias educacionais, deve-se pensar no gerúndio do verbo. Pensando os objetivos em termos de verbos, substantivos e gerúndios possibilita a escolha

de estratégias, conteúdos e instrumentos de avaliação eficazes e efetivos. A montagem da matriz deve se iniciar a partir da definição dos objetivos específicos da disciplina, curso ou conteúdo [Ferraz e Belhot 2010].

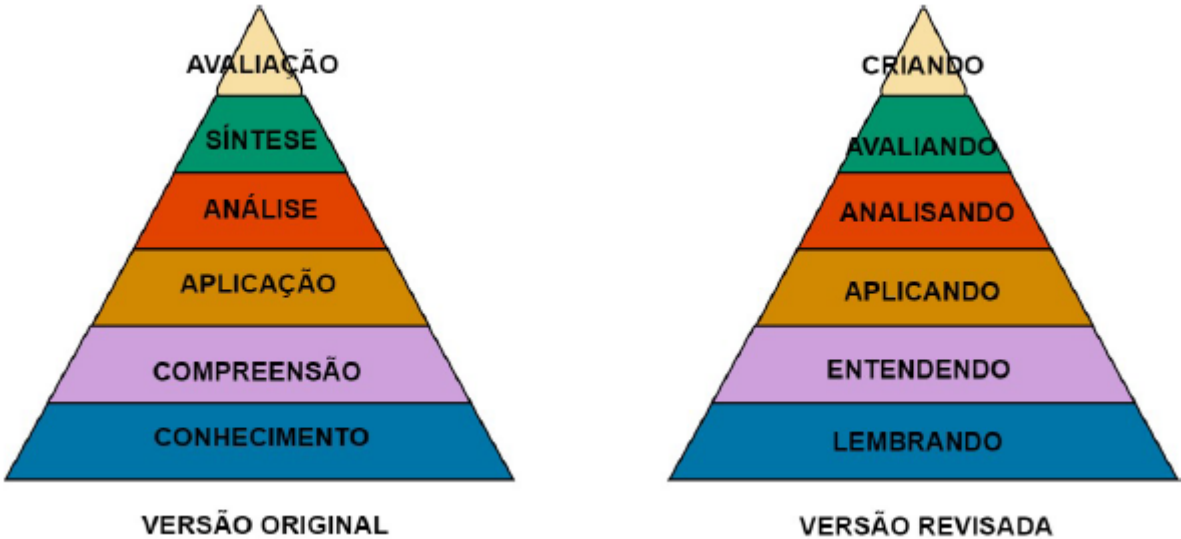
Figura 5 – Matriz Bidimensional

Dimensão conhecimento	Dimensão processo cognitivo					
	Lembrar	Entender	Aplicar	Analisar	Sintetizar	Criar
Efetivo/factual	Objetivo1					
Conceitual		Objetivo 2	Objetivo 2			
Procedural					Objetivo3	
Metacognitivo						
	Conhecimento		Competência		Habilidade	

Fonte: [ANDERSON et al., 2001]

Observe a figura 6, onde as habilidades cognitivas estão representadas. Na base da pirâmide está ilustrada a habilidade mais básicas e no topo, a mais avançada. Para que o aluno possa desenvolver uma certa habilidade de nível mais alto, ele deverá ter o domínio de habilidades de níveis inferiores a qual ele deseja atingir [Santos 2016].

Figura 6 – Matriz Bidimensional



Fonte: [ANDERSON et al., 2001]

O que torna a utilização da Taxonomia de Bloom um instrumento adequado para ser utilizado no ensino de terceiro grau é que ela, nos últimos anos, foi avaliada e atualizada considerando os avanços estratégicos e tecnológicos incorporados ao meio educacional [Ferraz e Belhot 2010].

2.5.1 Verbos RBT

Anderson e Krathwohl [Anderson e Krathwohl 2001] criaram uma lista de verbos para cada nível cognitivo dentro da RBT, de modo que tais verbos direcionam o modo como uma questão deve ser respondida [Peckham e McCalla 2012]. Baseados em [Ferraz e Belhot 2010], exemplos de verbos pertencentes a cada categoria da RBT são listados a seguir:

- **Lembrar:** enumerar, definir, descrever, identificar, denominar, listar, nomear, combinar, realçar, apontar, relembrar, recordar, relacionar, reproduzir, solucionar, declarar, distinguir, rotular, memorizar, ordenar e reconhecer;
- **Entender:** alterar, decodificar, defender, definir, descrever, distinguir, discriminar, estimar, dar exemplos, ilustrar, inferir, reformular, prever, classificar, discutir, identificar, reconhecer, redefinir, selecionar;
- **Aplicar:** aplicar, alterar, organizar, prever, produzir, usar, esboçar, escolher, escrever, praticar;
- **Analisar:** analisar, reduzir, classificar, comparar, contrastar, determinar, deduzir, diagramar, distinguir, diferenciar, identificar, ilustrar, apontar, inferir, subdividir, calcular;
- **Avaliar:** Avaliar, comparar, concluir, contrastar, criticar, justificar, relatar, resolver, validar, julgar;
- **Criar:** categorizar, combinar, compilar, compor, conceber, construir, criar, desenhar, elaborar, estabelecer, explicar formular.

2.6 Ambientes Virtuais de Aprendizagem

“Ambiente Virtual de Aprendizagem é um recurso tecnológico interativo com o objetivo de promover o diálogo entre professores e alunos, bem como fomentar a transmissão do conhecimento por meio de ferramentas apropriadas” (SILVA; FIGUEIREDO, 2012, p. 4). Assim, entende-se que um AVA é um local virtual onde são disponibilizadas ferramentas que permitem o acesso a um curso ou disciplina e também permite a interação entre os alunos, professores e monitores envolvidos no processo de ensino-aprendizagem.

O uso de AVA's, tem se tornado uma prática constante em educação presencial e a distância. Em geral, sua utilização não exige dos usuários um domínio mais aprofundado de informática, sendo necessárias apenas algumas horas ou poucos dias para que se aprenda a usar o ambiente.

Esses ambientes fornecem ao aluno um acesso mais flexível à educação, tendo a opção de gerenciar o seu horário e o seu local de estudo. Isso gera uma possibilidade de estudo para pessoas que por questões de distância geográfica, horários de trabalho ou por outras razões não puderam ou não podem cursar o ensino presencial (SILVA; FIGUEIREDO, 2012).

“Os AVAs têm por objetivo principal figurar como um espaço de construção do conhecimento por meio do desenvolvimento de atividades educativas, mediadas pelo uso de Tecnologias de Informação e Comunicação (TIC), valorizando a interação e o trabalho colaborativo” (MARTINS et al., 2009, p. 115).

Os AVAs podem conter vários tipos de usuários, no entanto, Almeida (2003) cita três que são muito comuns de serem encontrados:

- **Professores:** São os responsáveis pela disciplina, definindo como o ambiente será configurado.
- **Tutores:** Auxiliam os professores em relação ao acompanhamento dos alunos. Possuem alguns privilégios dentro do AVA, porém, esses privilégios ainda são limitados se comparados aos dos professores.
- **Alunos:** São os interessados no curso e realizam as tarefas das disciplinas as quais estão relacionados.

Existem muitas plataformas de AVAs, algumas pagas e outras gratuitas, diferenciando-se entre si por características variadas como nível de interatividade, usabilidade, capacidade colaborativa, dentre outras. Segundo Silva e Figueiredo (2012), os AVAs podem conter ferramentas que possibilitem a interação, interatividade, aspectos pedagógicos e administrativos como as mostradas na tabela 3:

Tabela 3 – Ferramentas para interatividade e interação

Exemplos	Categoria	Descrição
Correio Eletrônico	Comunicação/Interação	Usado para enviar e receber arquivos anexados às mensagens e se comunicar
Chat	Comunicação/Interação	Possibilita a comunicação de forma mais interativa e dinâmica.
Fórum	Comunicação/Interação	Ferramenta propícia para debates coletivos.

Continua na próxima página

Tabela 3 – Ferramentas para interatividade e interação

Exemplos	Categoria	Descrição
Lista de Discussão	Comunicação/Interação	Auxilia o processo de discussão entre os usuários inscritos em uma lista.
Mural	Comunicação/Interação	Possibilita que alunos e professores disponibilizem mensagens que podem ser de interesse para a turma. Normalmente são divulgação de links, convites para eventos, notícias rápidas, etc
Portfólio	Comunicação/gerenciamento	Auxilia na disponibilização dos trabalhos dos alunos e realização de comentários pelo professor e colegas da turma.
Anotações	Gerenciamento/comunicação	É uma ferramenta de gerenciamento de notas de aulas, observações, conclusão de assuntos, etc.
FAQ	Gerenciamento/comunicação	Auxilia o tutor/professor a responder as dúvidas mais frequentes, e com isso o aluno pode verificar se já existe resposta para a sua dúvida evitando o questionamento ao professor.
Perfil	Gerenciamento	Disponibiliza informações pessoais dos usuários.

Continua na próxima página

Tabela 3 – Ferramentas para interatividade e interação

Exemplos	Categoria	Descrição
Acompanhamento	Gerenciamento	Auxilia o professor no acompanhamento do aluno através dos relatórios que são gerados, com informações sobre histórico de acesso, notas, frequência por seção do ambiente visitado, histórico dos artigos, histórico de mensagens postadas para o fórum, participação em sessões de chat e mapas de interação entre os professores e alunos.
Avaliação(on-line)	Gerenciamento/comunicação	Está relacionada as avaliações que devem ser feitas pelos alunos e recursos para que possibilitem que o professor corrija as avaliações. Fornece ainda informações a respeito das notas, registro de quais avaliações o aluno realizou, tempo gasto para resposta e outros.

Como foi visto, os AVAs fornecem vários recursos aos usuários, entre esses recursos é comum encontrarmos: e-mails, listas abertas de mensagens, fóruns, portfólios, conferências, chats, wikis, blogs, quizzes, questionários, dentre outros.

Alguns possuem mais recursos que outros. A escolha de qual será utilizado deve ser feita mediante a análise detalhada de suas ferramentas, sua usabilidade e ganho educacional (FERRO, 2010).

Na tabela 4, encontram-se as características de algumas plataformas AVAs que mais apareceram na literatura durante a presente pesquisa e suas características são descritas baseadas na ideia de Gabardo et al. (2010), sendo o Moodle uma das plataformas gratuitas mais utilizadas.

Tabela 4 – Características de algumas plataformas de AVAs

Plataforma	Aprendizagem colaborativa	Interatividade	Multimídia	Usabilidade
Tel Educ	Grupos de discussão	Correio eletrônico, mural, portfólio, diário de bordo, chat, enquetes, fórum de discussão	Não apresenta	Alta - fácil de usar e bem explicativo
AulaNet	Propõe atividade colaborativa	Não apresenta	Não apresenta	Média - explica o uso, porém, apresenta problemas com barra de rolagem e textos com visibilidade parcial
Moodle	Fórum de discussão, Gestão de conteúdos, blogs, wikis	Vídeo-conferência e certificados digitais	Não apresenta	Alta - pois permite acesso de visitantes e oferece ferramenta para deficientes visuais

Esse conjunto de características fornecem uma visão sobre os recursos que normalmente compõem tais ambientes, tendo uma variação de um para o outro, mas ao mesmo tempo mantendo características em comum. Além disso, traz a ideia do que tais ferramentas possibilitam no apoio a educação.

2.7 Moodle

Moodle é um ambiente virtual de aprendizagem desenvolvido pelo australiano Martin Dougiamas. Seu nome é um acrônimo de **Modular Object-Oriented Dynamic Learning**, como mostra a figura 7. A modularidade e a orientação a objetos são duas praticas muito populares na área da engenharia de software. O Moodle também é modular, e este fato automaticamente o torna extensível, pois funciona a partir de um conjunto de módulos independentes, permitindo a personalização do ambiente.

No Brasil, esse AVA é bastante utilizado por universidades federais em seus cursos de Educação a distância, escolas em diversos níveis de ensino, departamentos de governo, organizações militares, companhias de petroleo e outros ³.

Figura 7 – Detalhamento da representação do nome do Moodle



Fonte: Página no Slideshare ⁴

Este AVA possui ampla variedade de recursos, permitindo que cursos sejam cadastrados, gerenciados e utilizados por seus usuários. O Moodle é utilizado em vários países, com grande quantidade de sites registrados e outros detalhes que podem ser vistos na figura 6. Nessa figura observamos as estatísticas com relação ao uso do Moodle ao redor do mundo:

³Disponível em <<https://docs.moodle.org/34/en/Usage>> Acessado em 19 de setembro de 2017

⁴Disponível em <<https://pt.slideshare.net/David.Melo-/moodle-arquitetura-e-recursos>> Acessado em 22 de setembro de 2017

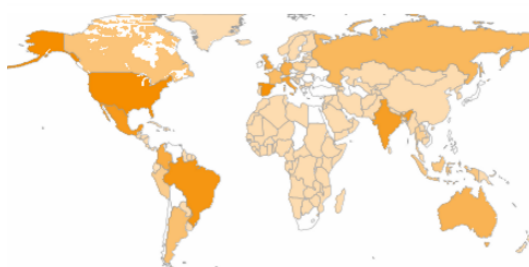
Figura 8 – Estatísticas gerais

Registered sites	90,598
Countries	233
Courses	14,107,330
Users	120,552,291
Enrolments	483,434,475
Forum posts	247,803,862
Resources	126,872,162
Quiz questions	701,123,562

Fonte: Página do Moodle⁵

Dentre os países que usam este ambiente, o Brasil está no Top 10 com relação a quantidade de sites registrados, como podemos ver na Figura 7.

Figura 9 – Países com maior número de registros



Country	Registrations
Estados Unidos da América	10,864
Espanha	7,999
Mexico	5,321
Brasil	5,315
Índia	4,109
Itália	3,858
Reino Unido da Grã-Bretanha e da Irlanda do Norte	3,664
Colômbia	3,044
Alemanha, República Federal da	2,810
Austrália	2,520

Fonte: Página do Moodle⁴

O ambiente Moodle é atualmente utilizado pela UFPB virtual em seus cursos de Educação a Distância.

Na plataforma é possível cadastrar cursos diretamente relacionados as suas respectivas disciplinas, ficando sob o controle do professor e dos tutores. Os alunos podem acessar o curso para realizar as atividades planejadas pelo professor, com o acompanha-

⁵Disponível em <<https://moodle.net/stats/>> Acessado em 19 de setembro de 2017

mento extra realizado pelos tutores que auxiliam os alunos nos fóruns de discussão, chats e trocas de mensagens (FERRO, 2010).

Segundo Ferro (2010), fica a cargo do professor definir quais ferramentas deseja configurar, possibilitando seu uso nas disciplinas. Como opção de ferramentas que podem ser configuradas pelos professores temos:

- **Blog:** Ambiente onde os usuários constroem seus próprios blogs;
- **Calendário:** Disponibiliza datas com as suas respectivas atividades;
- **Chats:** Possibilita a conversa em tempo real entre os usuários;
- **Escolha:** Possibilita a configuração de questionários que são posteriormente respondidos pelos alunos;
- **Fóruns:** Permite que tópicos sejam discutidos
- **Glossário:** Nesta ferramenta os alunos podem construir significado para palavras de forma colaborativa;
- **Recursos:** Local para postagem de arquivos que serão compartilhados com os alunos. Esses arquivos podem ser artigos, textos, apostilas e outros materiais didáticos;
- **Tarefas:** Mostra a listagem das tarefas que devem ser realizadas pelos alunos
- **Perfil:** Serve para os usuários descreverem seu perfil
- **Plano de curso:** Disponibiliza o plano da disciplina para os alunos

Além da ajuda proporcionada pelas ferramentas, os professores podem postar material didático, realizando o upload de arquivos (FERRO, 2010). É possível aprender mais sobre a funcionalidade e o histórico geral do Moodle no site⁵ do ambiente.

Podemos concluir que, o Moodle é um AVA amplamente adotado nos cursos a distância, possuindo recursos que podem facilmente ser estendidos para atender a necessidades específicas.

Neste capítulo, foi possível entender de forma detalhada os vários conceitos relevantes para o entendimento da construção da proposta do presente trabalho. No próximo capítulo iremos mostrar um pouco dos trabalhos que serviram como estímulo para este.

⁵Disponível em <http://docs.moodle.org/en/About_Moodle> Acessado em 05 de outubro de 2017

3 TRABALHOS RELACIONADOS

Neste capítulo são apresentados os principais trabalhos que serviram como inspiração, relatando exatamente quais pontos foram importantes para formar uma base para o presente trabalho.

Podemos dividir esse capítulo nas seguinte sequência mostrada abaixo:

- Apresentação do trabalho de [Carneiro 2018], que aplica a RBT no processamento dos dados a serem usados no presente trabalho
- Revisão sistemática sobre o estado da arte em Sistema de Recomendação, principalmente se for um sistema adaptativo na área de aprendizagem
- Análise, comparação e escolha de abordagens de modelos, métodos e técnicas que possam ser úteis para o presente trabalho

Durante a pesquisa foram encontrados vários trabalhos, porém, escolheu-se analisar detalhadamente apenas alguns, justamente por possuírem semelhanças com o que pretende-se solucionar no presente trabalho.

Nessa primeira parte trazemos a apresentação do trabalho de [Carneiro 2018], fazendo uso da RBT para realizar o processamento dos dados que serão usados nesse trabalho.

3.1 Trabalho que emprega a Taxonomia de Bloom e RBT na estruturação e categorização dos dados

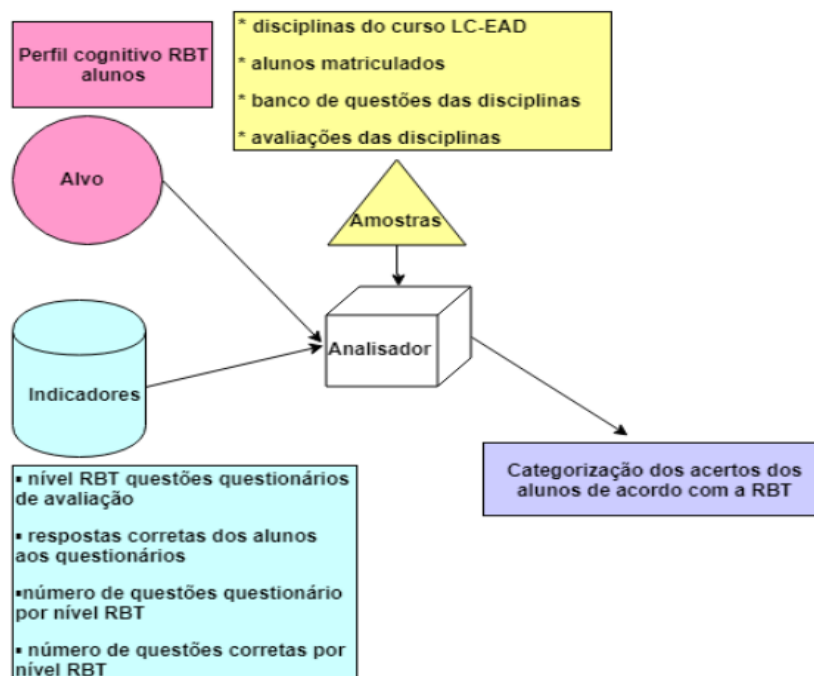
No Moodle, as questões não foram construídas com base na RBT, portanto, o trabalho de [Carneiro 2018] se propõe a resolver tal problema ao representar o perfil cognitivo do aluno de acordo com a RBT nos cursos do Moodle.

O trabalho elaborou o modelo mostrado na figura 10. Tal modelo, ajudou na estruturação, coleta e processamento das informações. O modelo proposto apresenta os seguintes elementos:

1. **Target:** Avaliação do perfil cognitivo do aluno de acordo com a RBT;
2. **Indicadores:** Nível RBT das questões de questionários de avaliação, respostas corretas dos alunos à essas questões, número de questões do questionário por nível RBT, número de questões corretas por nível RBT;
3. **Amostras:** Disciplinas do curso LC-EAD, alunos matriculados, banco de questões, avaliações das disciplinas;

4. Analisador: Descrito na seção 4.2.1.1

Figura 10 – Modelo Proposto - Fonte: [Carneiro 2018]



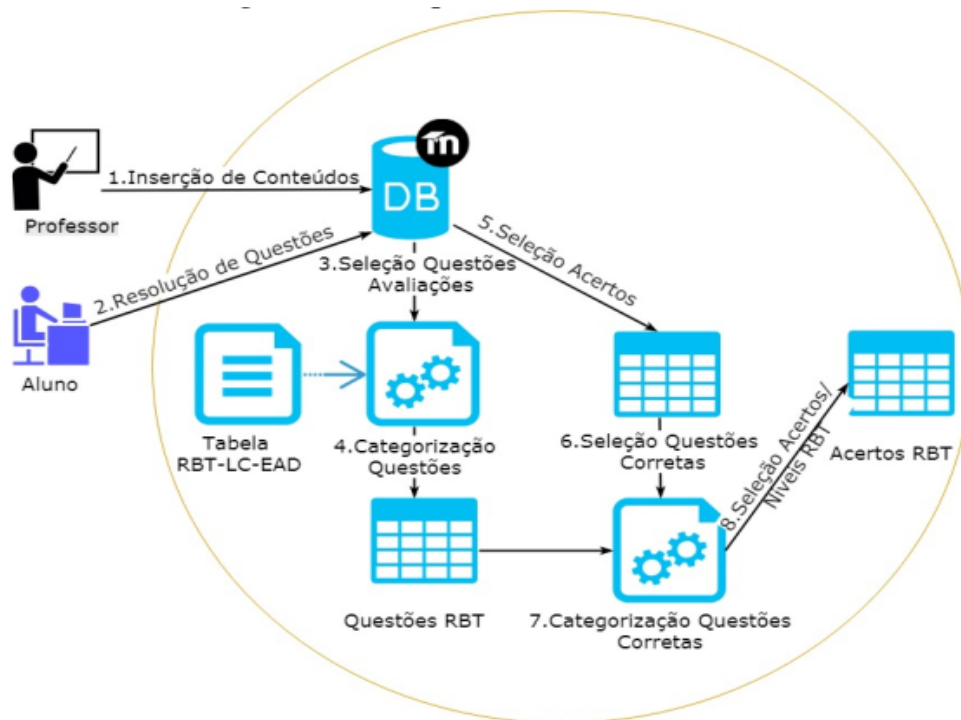
O trabalho também apresenta uma sugestão de [Souto 2003], que visa fazer a distinção dos dados, para os Sistemas Adaptativos de Ensino e Aprendizagem na Web:

- **Dados do aluno:** compreendem as características do mesmo e são representadas no modelo do aluno. Por exemplo, os registros dos alunos nos sistema.
- **Dados do ambiente:** São os dados que não se referem ao aluno, mas com o ambiente de ensino. Por exemplo, o registro do banco de questões.
- **Dados de uso do ambiente educacional:** são os dados referentes às interações do aluno com esse ambiente. Por exemplo, o estado das questões, se foram respondidas corretamente ou se foram resolvidas de maneira errada.

A descrição e detalhes sobre a estrutura de armazenamento do Moodle, e tudo o que é possível extrair em relação a dados do aluno, dados do ambiente e dados de uso do ambiente educacional são mostrados brevemente no trabalho de [Carneiro 2018] e em mais detalhes no próprio site do Moodle. Aqui focaremos apenas no que é importante para o entendimento do presente trabalho, que são as informações que temos e que podemos extrair, buscando ter uma visão geral de quais dados serão usados.

Com base no modelo proposto na figura 10, o analisador construído em [Carneiro 2018] é mostrado na figura 11:

Figura 11 – Arquitetura do Analisador - Fonte: [Carneiro 2018]



O analisador apresentado equivale ao componente analisador do modelo proposto na figura 10 e a figura 11 representa a arquitetura do analisador, que contém os personagens que interagem com o Moodle e o resultado dessa interação. Esse analisador consiste no processo de categorização das questões e acertos do aluno nos níveis RBT.

A figura 11 apresenta uma sequência de passos numerados que são detalhados abaixo:

1. Representa a preparação do ambiente de aprendizagem (inserção das questões de avaliações) por parte dos professores das disciplinas
2. Representa a interação dos alunos com o sistema (resposta às avaliações)
3. Representa a extração das questões utilizadas nas avaliações das disciplinas
4. Categorização das questões de acordo com os níveis RBT usando a lista RBT-LC-EAD
5. As questões que foram respondidas corretamente são extraídas do Moodle e serão inseridas na tabela Questões Corretas
6. As questões corretas são recuperadas
7. Essas questões serão categorizadas pelo processo de categorização de questões corretas que busca o nível das questões acertadas na tabela de questões RBT e as armazena na tabela Acertos RBT.

Para o presente trabalho ocorre uma mudança justamente nas etapas de 5 até 7. Para o passo 5 e 6, não serão extraídas/recuperadas apenas as questões respondidas corretamente, mas sim as corretas e as erradas. No passo 7, as questões continuarão sendo categorizadas através do uso da tabela de questões RBT, no entanto, serão consideradas tanto as questões respondidas corretamente quanto as respondidas erradamente e serão armazenadas na tabela Acertos-Erros-RBT.

Essa mudança se faz importante, pois, servirá como um complemento nos dados necessários para o módulo de eficiência do presente trabalho que será explicado em detalhes na seção 4.2.4.

Para entender como é possível recuperar as questões respondidas erradamente e corretamente no Moodle é preciso entender o funcionamento da question engine (mecanismo de perguntas) do Moodle, no que se refere a mudança de estados[Moodle 2016]. Existem 16 estados em que uma questão pode se encontrar. Cada estado de questão apresenta as propriedades que podem ser vistas na imagem abaixo. O estado que representa as questões finalizadas e acertadas se chama "gradedright" e o que representa questões finalizadas e erradas se chama "gradedwrong". Esses dois estados guiarão a seleção das questões de avaliação solucionadas corretamente e erradamente pelos alunos.

A figura 12 mostra os 16 estados possíveis para as questões.

Figura 12 – Propriedades que Representam os Estados das Questões - Fonte: Moodle

State	is_active	is_finished	is_graded	is_correct	is_partially_correct	is_incorrect	is_gave_up	is_commented
notstarted								
unprocessed								
todo	Y							
invalid	Y							
complete	Y							
needsgrading		Y						
finished		Y						
gaveup		Y					Y	
gradedwrong		Y	Y			Y		
gradedpartial		Y	Y		Y			
gradedright		Y	Y	Y				
manfinished		Y						Y
mangaveup		Y					Y	Y
mangrwrong		Y	Y			Y		Y
mangrpartial		Y	Y		Y			Y
mangrright		Y	Y	Y				Y

Com base no trabalho mostrado, foi possível concluir que o Moodle passa a ser um ambiente estruturado de acordo com a RBT. Em nenhum outro estudo a RBT é

utilizada como ferramenta para determinar o perfil de aprendizagem cognitivo do aluno em ambientes que não foram construídos de acordo com as diretrizes dessa taxonomia.

3.2 Revisão Sistemática em Sistemas de Recomendação

A revisão sistemática relacionada com a Sistemas de Recomendação foi realizada visando encontrar respostas para as seguintes questões de pesquisa:

- **QP1:** Como um sistema de recomendação pode ser estruturado em relação as informações de entrada disponíveis?
- **QP2:** Quais métodos e técnicas podem ser utilizados para gerar as recomendações baseado no modelo?
- **QP3:**Quais são as ações do sistema após ter a recomendação pronta para os usuários?

Essa revisão foi realizada em três fases (planejamento, realização e análise de resultados), de acordo com os seguintes critérios:

- Fase 1 - Seleção de estudos obtidos pela string de busca
 - estudos que contêm a string de busca pré-definida no título ou resumo
 - datados entre 2010 e 2018
- Fase 2 - Análise do título e resumo
 - Verificação rápida dos estudos, com o intuito de analisar se eles realmente cobrem aspectos da área de pesquisa. Consideramos apenas estudos primários. Isso significa que revisões informais ou sistemáticas não são consideradas para a próxima fase
- Fase 3 – Análise da Relevância
 - Verificação do texto completo, que pode ser recuperado na base. É analisado de modo a saber se o estudo contém informação suficiente para responder pelo menos duas das questões de pesquisa

Esses critérios foram utilizados para filtrar os estudos recuperados no seguinte mecanismo de busca: Google Acadêmico.

A string de pesquisa utilizada foi a seguinte:

((sistemas de recomendação OR recommender systems) AND (ensino a distância OR distance education) AND (ambiente virtual de aprendizagem OR virtual learning environment OR learning management system) AND (adaptive OR customized OR personalizado OR adaptativo OR customizado OR personalizado) AND (Coeficiente de Tanimoto OR Tanimoto Coefficient) AND (user based recommender OR recomendação baseada no usuário) AND (collaborative filtering OR filtragem colaborativa))

A avaliação de significância, sobre o objetivo da revisão sistemática, foi destinada a garantir que os artigos incluídos estivessem realmente focados em sistemas de recomendação na área aprendizagem adaptativa e contivessem detalhes suficientes para responder às questões de pesquisa pré-definidas.

Além da string de busca, alguns trabalhos também foram procurados de acordo com a necessidade, seja ao descobrir técnicas que não eram mostradas em alguns trabalhos, seja por motivos da boa didática em outros trabalhos, logo, acabou resultando no surgimento de alguns trabalhos interessantes que não foram retornados pela string de busca.

3.2.1 Resultados da Revisão em Sistemas de Recomendação

Os resultados da revisão são mostrados na tabela 5. Essa tabela mostra que 1010 artigos foram retornados na primeira fase, no entanto apenas 120 foram analisados. Já na segunda fase o número de trabalhos caiu para 49 e na terceira fase ficamos com 7.

Figura 13 – Tabela da Revisão Sistemática em SR

Base de Dados	Número de Artigos Selecionados		
	Fase 1	Fase 2	Fase 3
Google Acadêmico	120 (981)	49	7

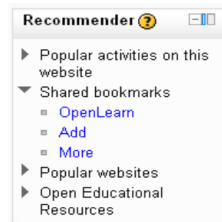
A seguir, apresentamos os trabalhos, e logo em seguida, iremos expor como eles ajudaram na obtenção das respostas para as perguntas QP1, QP2 e QP3.

No Moodle, temos por exemplo um plugin denominado Recommender [Moodle 2013], disponível no ambiente em forma de bloco, como mostra a figura 14, e ele oferece alguns serviços de recomendação como exemplifica a lista abaixo:

- **Atividades mais populares:** lista 10 recursos e atividades mais vistos/atualizados no curso em que o bloco está sendo visualizado no momento;
- **Cursos mais populares:** lista os 3 cursos com maior número de alunos matriculados em comum com o curso em que o bloco está sendo visualizado no momento;

- **Bookmarks compartilhados:** possibilita aos professores fazerem suas próprias recomendações, adicionando qualquer link ao bloco.

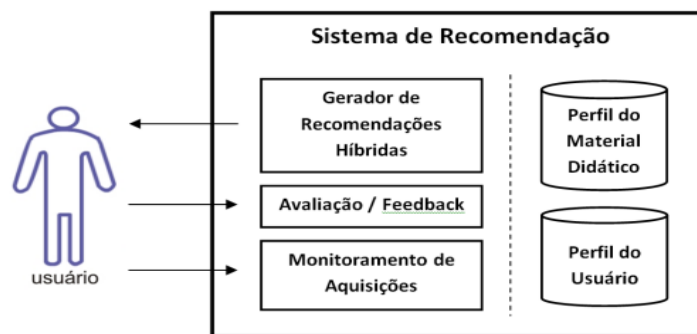
Figura 14 – Plugin de bloco Recommender - Fonte: [Moodle 2013]



Apesar de não utilizar abordagens como filtragem colaborativa e filtragem baseada em conteúdo, o plugin Recommender é um exemplo de utilização do conceito de recomendação de itens no contexto educacional.

Já o trabalho de [Ferro 2010], traz a proposta de um modelo de sistema de recomendação de materiais didáticos para AVAs. Para solucionar esse problema, utilizou-se uma técnica de geração de recomendação híbrida, que nesse caso é uma combinação da recomendação não personalizada [Schafer et al. 1999 apud Ferro et al. 2011], recomendação baseada no conteúdo e recomendação baseada em filtragem colaborativa, como mostra a figura 15.

Figura 15 – Arquitetura do gerador de recomendações híbridas



Fonte: FERRO (2010)

De um modo geral, o processo de geração de recomendações realizado consiste nestas quatro etapas mostradas na lista abaixo:

1. Geração de uma lista de materiais didáticos ao aplicar filtragem colaborativa. A posição de cada item contido nesta lista é denominada posição de premeditação.
2. Geração de uma lista ordenada, estilo um ranking dos materiais da lista gerada na etapa anterior. Tal ordenação é feita a partir do Grau de Utilidade da técnica de re-

comendação baseada no conteúdo. A posição de cada item dessa lista é denominada posição da utilidade.

3. Geração de uma lista ordenada, estilo um ranking dos materiais da lista gerada em na primeira etapa. Sua ordenação é feita através da procura dos materiais didáticos mais requisitados por outros usuários (técnica de recomendação não personalizada). A posição de cada item dessa lista é denominada Posição do Quantitativo.
4. Cálculo dos Graus de Utilidade Híbrida, a partir das listas geradas nas três etapas anteriores, cuja equação é mostrada no figura 16.

Figura 16 – Equação para cálculo dos graus de utilidade híbrida

$$uh(item) = \frac{1}{Posição\ da\ premeditação(item)} + \frac{1}{Posição\ da\ utilidade(item)} + \frac{1}{Posição\ do\ Quantitativo\ (item)}$$

Fonte: FERRO (2010)

Outro ponto importante é que, tal trabalho teve um protótipo desenvolvido no Moodle que implementou o modelo proposto.

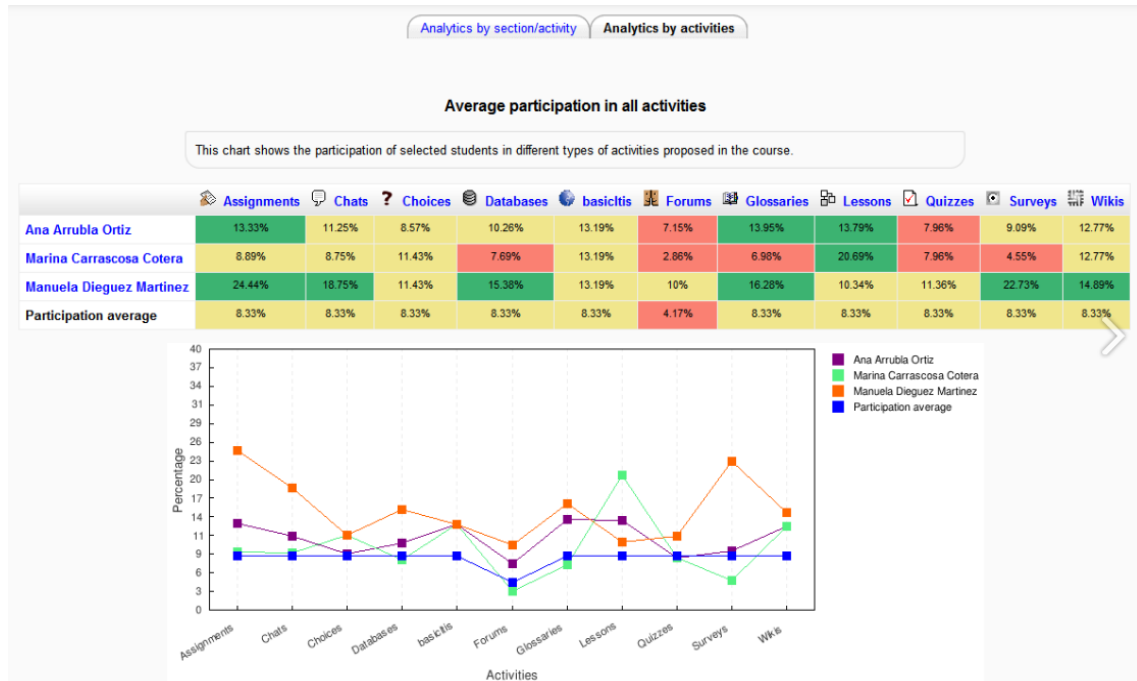
[Hassan e Hamada 2016] investigam o fato de que os Sistemas de Recomendação Educacionais geralmente utilizam um único atributo de avaliação do usuário para prever o seu grau de interesse no objeto de aprendizagem. Contudo, as preferências do usuário podem depender de mais de um tipo de atributo, por exemplo, autor, nível de dificuldade e muitas outras possibilidades. Considerando isto, os autores criam um Sistema de Recomendação multi-critério, que modela e estima as preferências do usuário usando um vetor de atributos de avaliação, que contempla o valor de avaliação para cada atributo do objeto de aprendizagem, ao invés de um único atributo. Além disso, a técnica utiliza uma Rede Neural Artificial para a tarefa de aprender a recomendar. Os autores mencionam que etapa futura do trabalho é a integração do recomendador a um AVA.

Outro trabalho que serve como referência foi o plugin de bloco *Analytics and Recommendations* que é disponibilizado para a plataforma Moodle. Esse plugin tem como objetivo mostrar através de elementos visuais a participação dos alunos em cada atividade do curso e mostrar recomendações sobre quais atividades devem trabalhar para melhorar sua nota final, baseado na participação em cada tipo de atividade ⁶. A figura 17 mostra um exemplo do plugin em funcionamento, no entanto, essa figura não expõe todo o seu poder, pois representa apenas uma parte do que ele é capaz de realizar.

⁶Disponível em <https://moodle.org/plugins/block_analytics_recommendations> Acesso em 11 de novembro de 2017

Este plugin teve impacto para o presente trabalho no processo de identificar se existiam mais dados além das obtidas em [Carneiro 2018], para a modelagem do recomendador, além de fornecer uma ideia de como esses dados podem ser explorados.

Figura 17 – Plugin de bloco Analytics and Recommendations em funcionamento



Fonte: Página do Moodle⁶

Para obter a resposta para a QP1, vamos considerar que em alguns trabalhos analisados durante a pesquisa, foi possível perceber que a entrada para um sistema de recomendação pode ser estruturada através de matrizes, com linhas e colunas referenciando objetos de interesse e o conteúdo da matriz sendo a relação entre ambos os objetos, porém nem sempre isso normalmente fica claro na maioria dos trabalhos. A ideia inerente as matrizes se encaixam bem para a estruturação dos dados que serão extraídos das tabelas construídas no trabalho de [Carneiro 2018] que estão no banco de dados da instância do Moodle, visando montar a entrada para o recomendador proposto no presente trabalho.

Para responder a QP2, vamos levar em consideração o trabalho de [Hassan e Hamada 2016], juntamente com o trabalho de [Ferro 2010], ambos mostram várias técnicas, métodos e como eles interagem entre si para gerar a construção desejada. A maioria dos outros trabalhos também apresentam normalmente os mesmos conceitos, diferindo apenas na abordagem, na aplicação e na adição de outros conceitos. Dentre as técnicas usadas nos trabalhos, podemos destacar a filtragem colaborativa baseada no usuário, que internamente realiza cálculos de vizinhança e cálculos de similaridades com base nas informações passadas como entrada e essa será a técnica usada na construção de uma parte do modelo de recomendador proposto neste trabalho.

Para a QP3 não existe uma regra, depende do que cada sistema se propõe a fazer com a recomendação, porém, a maioria dos trabalhos passam para o usuário alvo da recomendação os elementos a serem recomendados, o que também é o caso deste trabalho.

Como principal inspiração para a definição da arquitetura do SR proposto neste trabalho e para a identificação de quais etapas do processo de recomendação seriam necessárias usar no modelo de recomendação definido, podemos destacar o modelo de arquitetura proposto e as etapas a serem usadas no processo de recomendação de [Ferro 2010].

4 DEFINIÇÃO DA PROPOSTA DE MODELAGEM E IMPLEMENTAÇÃO

O modelo de SR aqui proposto, destina-se a gerar recomendações de questões para os alunos, buscando melhorar seu aprendizado. Este SR é híbrido, pois mistura a técnica de filtragem colaborativa baseada no usuário com um critério de eficiência que será mostrado adiante neste capítulo. A etapa precedente que fornece os dados usados no recomendador, vem da modelagem analítica para representar o perfil cognitivo do aluno de acordo com a RBT construído do trabalho de [Carneiro 2018].

4.1 Metodologia

Metodologicamente o trabalho foi dividido em duas etapas. A primeira é a fase exploratória e de revisão da literatura, e a segunda é a de projeto, construção e análise.

A primeira etapa desta pesquisa tem sua ordem cronológica mostrada na lista abaixo e é constituída dos seguintes passos:

1. **Desenvolvimento de uma revisão sistemática da literatura:** Essa etapa visou a investigação/identificação do estado da arte no domínio de aprendizagem adaptativa em AVAs
2. **Estudo do trabalho de [Carneiro 2018]:** Essa etapa visou investigar/identificar como usar o trabalho fase inicial do presente trabalho, e também verificar uma possível evolução na construção.
3. **Revisão da literatura em Sistemas de Recomendação:** Essa etapa visou investigar/identificar como os conceitos inerentes a sistemas de recomendações podem ser usados
4. **Investigação do estado da arte no domínio de AVAs:** Investigação dos AVAs existentes em relação as funcionalidades disponibilizadas, o quão adaptáveis cada um pode ser e quais são mais usados.
5. **Estudo do Moodle:** Visou determinar as características do ambiente Moodle que possibilitam a modelagem do aluno através das abordagens escolhidas e também dos dados disponíveis sobre a trajetória de aprendizagem do aluno do curso LC-EAD nesse ambiente
6. **Análise de sistemas que usam SR:** Foram pesquisados alguns sistemas que usam SR e foram inspecionados em relação ao seu funcionamento. Dentre os inspecionados temos Amazon e Netflix.

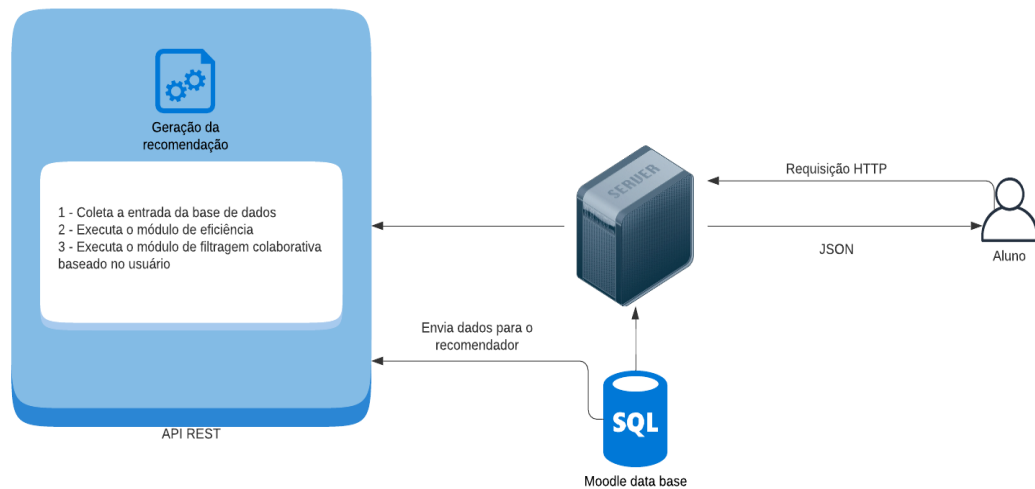
A segunda fase desta pesquisa tem sua ordem cronológica mostrada na lista abaixo. Tal etapa é constituída dos seguintes passos:

1. **Proposta de um modelo arquitetural de recomendação da aplicação:** Propõe um modelo arquitetural que serve como modelo para implementação, sendo independente do AVA e das ferramentas a serem usadas. Tal modelo integra o módulo RBT, o módulo de filtragem colaborativa e o módulo de verificação de quantidade e eficiência e no final visa sua disponibilização na forma de um serviço, o que possibilita sua utilização em diversas aplicações, não se limitando apenas ao Moodle.
2. **Módulo RBT:** Realiza a extração das informações necessárias para a modelagem e construção do módulo RBT e é composto de várias etapas. Essa parte foi literalmente extraídas do trabalho de [Carneiro 2018], de onde é retirado todo o módulo RBT, tendo apenas como acréscimo em relação ao trabalho citado, a coleta de informações das questões que o aluno errou, pois essa informação será útil no terceiro e último módulo do recomendador que será mostrado mais a frente.
3. **Módulo de verificação de quantidade e eficiência dos alunos por nível:** Propõe um módulo que compõe uma das etapas da recomendação, funcionando com base em um critério de eficiência que aplica uma filtragem para eliminar elementos da entrada.
4. **Módulo de filtragem colaborativa baseada no usuário:** Propõe outro módulo do recomendador, tendo como essência a técnica de recomendação de filtragem colaborativa baseada no usuário mostrada no capítulo de fundamentação teórica.
5. **Discussão e análise de resultados:** Para essa etapa foi planejada uma análise experimental, visando validar o comportamento do recomendador com base em instâncias de entrada do curso LC-EAD.

4.2 Modelo Proposto

A arquitetura para o modelo proposto é mostrado na figura 18 e é composto por três módulos, cada um com um papel bem definido na representação do recomendador.

Figura 18 – Arquitetura geral do recomendador



4.2.1 Módulo RBT

O desenvolvimento do módulo RBT se baseou no trabalho de [Carneiro 2018], realizando apenas uma melhoria em relação ao que tínhamos no trabalho citado, tal melhoria foi para passar a considerar na construção que será mostrada adiante, as questões que o aluno respondeu e errou, ao invés de considerar apenas as que ele acertou, pois isto é primordial para o módulo de eficiência que será mostrado.

O módulo RBT cognitivo do aluno, foi elaborado com base no modelo e arquitetura mostrados na seção 3.1, que expõe as soluções adotadas no trabalho de [Carneiro 2018], e sabemos que tal trabalho foi usado como etapa precedente para o presente trabalho. Tal modelo ajudou na coleta, estruturação e processamento das informações. O módulo proposto tem seus passos mostrados na lista abaixo:

1. **Análise manual dos verbos:** Nessa etapa foi realizada a análise manual dos verbos RBT do curso LC-EAD, a fim de identificar a ocorrência dos verbos da lista RBT nas questões de avaliações das disciplinas do curso em questão.
2. **Construção da lista de verbos:** Construção da lista de verbos RBT-LC-EAD contendo os verbos RBT identificados no passo anterior, realizada manualmente através do resultado do passo anterior
3. **Projeto da base de dados do modelo:** Especifica os dados necessários para a representação do perfil do aluno
4. **Desenvolvimento da base de dados do modelo:** realizada em um primeiro momento, através da importação de quatro disciplinas, bem como dos alunos participantes das mesmas, do curso LC-EAD, que foram disponibilizadas como fonte de

análise para esta pesquisa, e posteriormente através da criação das tabelas de acordo com o projeto realizado no passo anterior

5. **Formatação dos dados:** Formata os dados obtidos para armazenamento no modelo dimensional apresentado neste capítulo
6. **Scripts de extração dos dados do Moodle:** Script para alimentar as tabelas que formam o modelo do aluno construído.
7. **Categorização de questões RBT na fase exploratória:** esse passo serviu como base para a observação do perfil cognitivo do aluno, pois para essa observação, as respostas do aluno às questões categorizadas foram levadas em consideração
8. **Extração dos acertos e erros dos alunos:** Visa identificar questões de avaliações acertadas ou erradas pelo aluno.
9. **Categorização RBT dos acertos dos alunos:** visa identificar os níveis RBT equivalentes das questões de avaliações acertadas pelo aluno.

Tais etapas foram literalmente extraídas do trabalho de [Carneiro 2018], de onde é retirado todo o módulo RBT a ser implementado, tendo apenas como acréscimo em relação ao trabalho citado, a coleta de informações das questões erradas no nível 8 da lista mostrada acima. O resultado desse módulo, é passado como entrada para o módulo de verificação de eficiência dos alunos por nível.

Antes de iniciar a explicação dos próximos dois módulos, é importante ter em mente que os dados extraídos nesse módulo que servem como entrada para os próximos dois, seguem os seguintes critérios mostrados abaixo:

- Apenas questões relativas ao nível do aluno alvo devem ser usados
- Alunos em níveis inferiores ao aluno alvo na RBT serão desconsiderados, pois, esses não podem ajudar alunos de níveis superiores a subir níveis na RBT. Não faz sentido considerar quem ainda não está no nível avaliado. Por exemplo, podem ser acertos por sorte e isso não define o nível de conhecimento do aluno.
- Para os alunos de níveis superiores na RBT em relação ao aluno alvo da recomendação, serão consideradas apenas as questões de quando os alunos geradores estavam nesse nível. O motivo para desconsiderar as questões dos alunos geradores após eles terem subido do nível que está em avaliação, é que as questões de níveis mais baixos teoricamente se tornam mais fáceis a medida que se está em um nível mais alto e o que desejamos obter é o comportamento do indivíduo quando ele pertencia ao nível em análise, ou seja, os alunos geradores e o alvo devem se parecer considerando uma equivalência de nível.

4.2.2 Módulo de verificação de eficiência dos alunos por nível

Pensando na eficiência como uma maneira de medir se algo está sucedendo com um certo nível de excelência, então, para esse módulo, propomos a nossa própria medida de eficiência, que busca eliminar quem acertou muitas questões, mas em contra partida, também errou muitas. A ideia inerente a esse módulo é descrita em mais detalhes e em etapas na lista abaixo:

1. **Montagem da matriz de acertos/erros RBT em um nível específico:** Estrutura a matriz que contém todos os alunos, tais alunos são representados pelas linhas e as questões são representadas pelas colunas. O conteúdo das matrizes são os respectivos acertos que são marcados com 1 para questões respondidas corretamente, 0 para questões não respondidas e -1 para questões que foram respondidas e não acertadas.
2. **Construção** O cálculo da eficiência se baseia na fórmula abaixo. Caso a eficiência esteja abaixo de 0.7, o aluno também passa a ser desconsiderado como gerador de recomendações para o aluno alvo. Esse resultado sempre ficará entre 0 e 1, quanto mais próximo de 1, mais eficiente é o aluno e quanto mais perto de 0, menos eficiente ele é. Vale ressaltar que as questões previamente respondidas de forma errada podem ser refeitas, e esse fato poderia ser usado para evoluir o cálculo da eficiência em um trabalho futuro.

$$eficiencia = \frac{qtdQuestoesAcertadas}{qtdQuestoesRespondidas} \quad (6)$$

Para a nossa implementação, apenas alunos com eficiência superior a 0.7 são mantidos na lista de usuários que serão usados para gerar recomendação ao aluno alvo. O resultado desse módulo é a matriz montada nesta etapa, porém, com a eliminação dos que foram considerados ineficientes. Esse resultado é passado como entrada para o próximo e último módulo do recomendador chamado de módulo de filtragem colaborativa baseada no usuário.

4.2.3 Módulo de filtragem colaborativa baseada no usuário

De acordo com [Lamm 2013], na filtragem colaborativa, existem alguns algoritmos que utilizam dados booleanos como entrada. É justamente baseado nessa ideia booleana que vamos construir a entrada, pois se adequa perfeitamente ao modelo de informação que temos em relação as questões.

1. **Montagem das matrizes de acertos RBT de um nível específico:** Estrutura a matriz que contém todos os alunos. Tais alunos são representados pelas linhas e as

questões são representadas pelas colunas. O conteúdo das matrizes são os respectivos acertos que são marcados com 1 para questões respondidas corretamente, 0 para questões não respondidas

2. **Construção do algoritmo de recomendação:** O algoritmo é contruído baseado em filtragem colaborativa, que no caso desse trabalho é baseada no usuário, que internamente usa o *Coefficiente de Tanimoto* para o cálculo das similaridades entre os alunos através de interceções e em seguida calcula a vizinhança usando *threshold*. O algoritmo consome as informações definidas no passo anterior e realiza o processamento. Nessa construção, foi usada a biblioteca *Apache Mahout*.
3. **Montagem da resposta:** Como resultado do processamento realizado pelo algoritmo, temos uma resposta estruturada como um array contendo as questões a serem recomendadas em ordem de prioridade para o usuário alvo em relação a todos os usuários selecionados na vizinhança.

Um exemplo de como seria a representação dessa matriz de entrada é mostrado abaixo:

Tabela 5 – Questões resolvidas e acertadas por alunos

	Q1	Q2	Q3	Q4	Q5	Q6
Aluno 1	1	0	1	1	0	0
Aluno 2	1	1	0	0	1	1
Aluno 3	0	0	1	1	0	1
Aluno 4	1	0	0	0	0	0
Aluno 5	1	1	0	0	1	0

A flag 1 nos campos da tabela indicam que a questão foi resolvida e acertada. Não faz sentido pegar questões respondidas e não acertadas nesse ponto, pois queremos calcular a similaridade entre os alunos, para quando eles sabem o que respondem, ou seja, representa melhor o seu real conhecimento, logo, a similaridade identifica quem tem conhecimentos mais parecidos em um dado nível.

Como resultado, teremos uma lista de questões a serem recomendads em ordem de prioridade, sendo essa prioridade definida como a contagem da quantidade de questões que mais aparecem como possível alvo de recomendação.

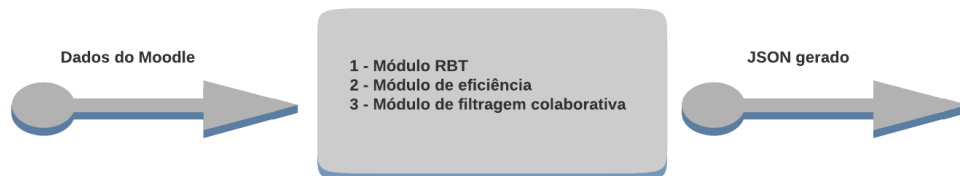
4.2.4 Integração dos Módulos

Após as etapas anteriores, o sistema terá realizado todo o necessário para ter a recomendação gerada. A recomendação é o resultado final obtido através da atividade

individual, sequencial e de colaboração exercida por cada módulo, indicando ao alvo as questões que mais se adequam ao seu perfil ordenadas por prioridade.

A arquitetura e a sequência de execução dos módulos é mostrado na figura 19 que aqui chamamos arquitetura do recomendador.

Figura 19 – Arquitetura do recomendador após a integração



4.3 Desenvolvimento

4.3.1 Tecnologias Usadas

Várias tecnologias foram usadas, mas vamos dividi-las para explicar em que parte do trabalho cada uma foi aplicada:

- O servidor local usado para instalação e uso do Moodle foi o Web Server Apache
- Para o módulo RBT foi utilizado MySQL para armazenamento/estruturação/construção das entidades
- Para construção do módulo de filtragem colaborativa baseada no usuário foi usado java(spring boot e JPA) em conjunto com a biblioteca Apache Mahout. Para a construção do módulo de eficiência e quantidade foi usado apenas a linguagemn Java. Além disso, para gerar o resultado final, que é a disponibilização da recomendação como um serviço REST, foi utilizado alguns elementos da família Spring que é um framework usado no Java.

4.3.2 Apache Mahout

Mahout surgiu em 2008 como um subprojeto do projeto Lucene do Apache. Lucene fornece implementações avançadas de técnicas de pesquisa, mineração de texto e recuperação de informações. Esses conceitos citados anteriormente são adjacentes à técnicas de aprendizado de máquina, como por exemplo agrupamento e, até certo ponto, classificação. Como resultado, alguns dos trabalhos dos membros de Lucene foram transferidos para o Mahout que era considerado um subprojeto. O Mahout também absorveu o projeto de filtragem colaborativa do Taste. A partir de abril de 2010, a Mahout tornou-se um

projeto individual da Apache, e obteve um novo logotipo. Uma grande porção do trabalho do Mahout nessa transição não se resumiu apenas na implementação desses algoritmos convencionalmente, de forma eficiente e escalável, mas também na conversão de alguns desses algoritmos para trabalhar em escala no Hadoop (OWEN et al., 2012).

O mahout possui três características importantes. A primeira delas é que é uma **biblioteca de aprendizado de máquinas de código aberto do Apache**. Os algoritmos que ele implementa caem na área de aprendizagem de máquina ou da inteligência coletiva, o que poderia significar muitas coisas, mas para Mahout significa principalmente motores recomendadores (filtragem colaborativa), agrupamento e classificação. A segunda delas é a **escalabilidade**. Mahout é uma boa ferramenta quando a coleção de dados a serem processados é muito grande, e em alguns casos também é muito grande para uma única máquina. As implementações que o tornam escalável são escritas em Java, e algumas partes são construídas com o projeto de computação distribuída Hadoop da Apache. A terceira é que é uma **biblioteca Java**. Essa biblioteca fornece um quadro de ferramentas destinadas a serem usadas e adaptadas pelos desenvolvedores (OWEN et al., 2012).

O Mahout é um projeto que se concentra em três áreas-chave da aprendizagem de máquinas no momento, que são: **motores recomendadores (filtragem colaborativa), agrupamento e classificação**. Isso não é tudo que existe no Mahout, mas são os temas mais proeminentes (OWEN et al., 2012).

Das três, a única usada na implementação foi a parte de **motores de recomendação**, que é uma das técnicas de aprendizado de máquina das mais usadas nos dias atuais e parte escolhida e aplicada na implementação foi a filtragem colaborativa oferecida pelo Mahout, que realiza o cálculo da similaridade e das respectivas previsões. O motivo para termos feito a escolha de usar essa parte da biblioteca, é que tais técnicas são muito usadas na construção de recomendadores, como foi mostrado no capítulo 3 do presente trabalho e também pode ser observado na literatura.

5 Resultados

Nesta seção, serão apresentados os seguintes resultados:

- Resultado do processo de análise do perfil cognitivo do aluno que puderam ser obtidos através do modelo proposto feito em [Carneiro 2018], que corresponde ao módulo RBT. Tais resultados serão mostrados aqui objetivando pontuar detalhes importantes para o presente trabalho
- Resultado do processo de recomendação gerado pelo recomendador implementado, mostrando diferentes casos com exemplos

5.1 Estudo Exploratório

O cenário de análise de resultados dessa pesquisa se divide em duas partes:

- **Resultados para o módulo RBT:** Análise, comentários, alterações e conclusões sobre os resultados obtidos no trabalho de [Carneiro 2018] que vão servir para este trabalho.
- **Resultados para o módulo de filtragem colaborativa baseada no usuário e eficiência:** Análise, comentários e conclusões a respeito das recomendações geradas por tais módulos, de modo a saber se os resultados obtidos estão de acordo com o esperado após ser aplicado em diferentes cenários

É importante ressaltar que os módulos foram implementados e testados separadamente, visando garantir que cada módulo funciona separadamente. Após isso, todos foram integrados e testados em conjunto.

Para análise do modelo, consideramos os dados gerados em quatro turmas de duas disciplinas do curso de Licenciatura em Computação (LC-EAD) da UFPB, tendo como base os semestres 2017.1, 2017.2 e 2018.1. Cada uma dessas turmas são codificadas conforme a identificação:

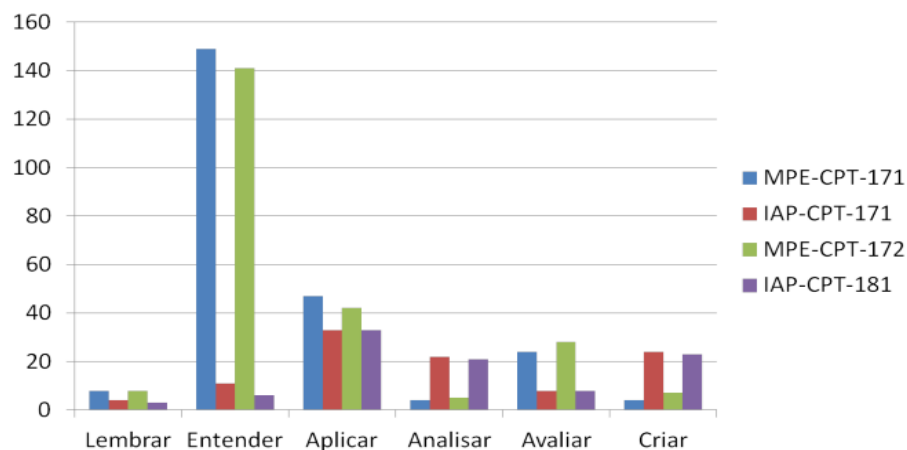
- MPE-CPT-171 - Matemática Elementar, semestre 2017.1
- MPE-CPT-172 - Matemática Elementar, semestre 2017.2
- IAP-CPT-171 - Introdução a Programação, semestre 2017.1
- IAP-CPT-181 - Introdução a Programação, semestre 2018.1

A princípio, as disciplinas selecionadas para o estudo de caso eram as duas de Introdução à Programação e as duas de Matemática Elementar. É importante ponderar que as disciplinas de Matemática Elementar, dos períodos 2017.1 e 2017.2, em conjunto com a disciplina de Introdução à Programação, do período 2017.1 contém dados equivalentes a 1(um) semestre letivo completo, porém, na disciplina de IAP-CPT-2018.1, ressalta-se que temos apenas dados do primeiro bimestre do curso, pois a disciplina estava em execução no momento em que a base de dados desta disciplina foi obtida para esta pesquisa.

5.1.1 Resultados módulo RBT

No trabalho de [Carneiro 2018] temos alguns resultados importantes e que precisam ser conhecidos para um melhor entendimento do presente trabalho. Tais resultados são mostrados de forma resumida nas imagens abaixo. Caso seja desejado compreender mais detalhes, o trabalho citado anteriormente traz uma análise mais minuciosa.

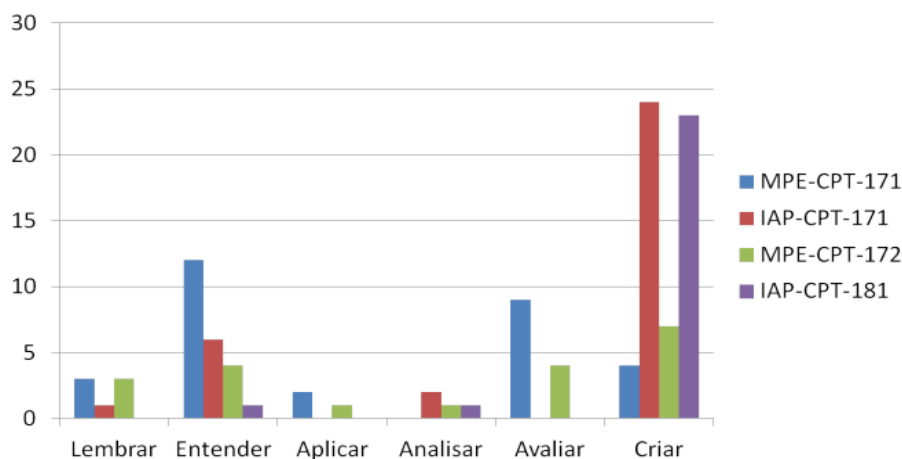
Figura 20 – Distribuição de questões RBT por disciplina



Da figura 20 é possível observar que existem questões em todos os níveis RBT para as disciplinas com uma variação entre 0 e 160 questões por nível RBT.

Vemos que da distribuição das questões classificadas segundo a RBT da figura 20 apenas a quantidade mostrada na figura 21 foi utilizada em avaliações.

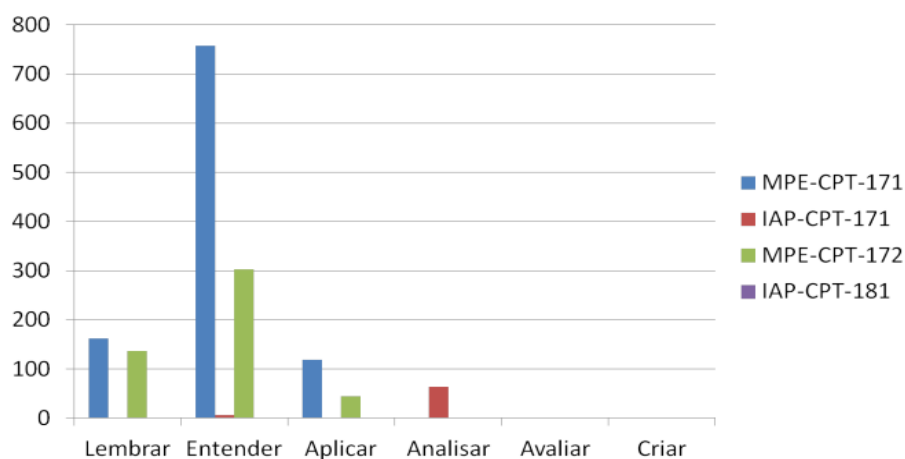
Figura 21 – Distribuição de questões RBT por disciplina usadas nas avaliações



Na figura 21 vemos que existem disciplinas sem questões resolvidas em um determinado nível RBT. Um ponto importante a se perceber é que a quantidade total de questões RBT nas avaliações variam de 0 até 30. Isso significa que uma quantidade pequena de questões categorizadas de acordo com a RBT foram inseridas nas avaliações.

Das questões que foram utilizadas nas avaliações dos cursos, a contagem das que foram acertadas pelos alunos em cada nível RBT por curso é mostrado na figura 22.

Figura 22 – Contagem de questões RBT por disciplina que foram acertadas nas avaliações



Essa análise possibilitou a escolha a respeito de quais dados serão usados na criação das recomendações, e como pode ser observado, o curso MPE-CPT-171, período 17.1 foi escolhido por ser o que tem mais representação de questões corretas, por nível, o que pode ajudar a diminuir o problema relativo a esparsidade. Com relação ao nível RBT, o escolhido é o nível Entender, possuindo um total de 12 questões usadas em avaliações.

5.1.2 Resultados do Recomendador

Uma forma eficaz de se avaliar sistemas de recomendação é através da comparação das predições realizadas com os respectivos resultados reais de utilizadores para as instâncias preditas. A obtenção de medidas para medir o desempenho de um sistema de recomendação antes da utilização é fundamental para verificar se as recomendações se enquadram no objetivo da solução [Martins 2016].

Aqui testaremos os resultados do módulo de filtragem colaborativa baseado no usuário e de eficiência para diferentes cenários, mas tendo como foco o principal e mais complexo módulo que é o de filtragem colaborativa baseada no usuário, ou seja, o recolhimento de dados do módulo RBT já vão ter sido executados nesse ponto. Para nosso trabalho, não temos notas na entrada, por isso não temos como realizar o testes mais comuns que fazem predição. O que temos são dados booleanos, ou seja, se a questão foi respondida e acertada ou não.

Serão mostrados cenários que contém redução de alunos, objetivando tornar a análise mais didática, evitando a apresentação de uma tabela imensa. Por exemplo, deixar de expor todos alunos que serão descartados durante o processo, por não serem adequados para geração de recomendação para aluno alvo, não impactaria no resultado final e no processo de entendimento do funcionamento do recomendador.

5.1.2.1 Cenário 1 - Threshold muito baixo

Na figura 23 é mostrada uma instância reduzida, como no módulo de eficiência não houve nenhum aluno sendo eliminado, então para esse primeiro cenário, tal etapa será suprimida. Para esse cenário o aluno 9 é o alvo das recomendações.

Figura 23 – Entrada do módulo de filtragem colaborativa baseado no usuário

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
Aluno 9												
Aluno 15												
Aluno 30												
Aluno 12												
Aluno 5												
Aluno 3												
Aluno 20												
Aluno 27												

Essa entrada foi representada na imagem 23 como uma tabela para fins didáticos. As células marcadas na cor verde indicam questões respondidas e acertadas e as células sem preenchimento de cor representam as questões não feitas.

O primeiro passo é realizar o cálculo do Coeficiente de Tanimoto para saber a similaridade entre o Aluno 9 e o restante dos alunos. A imagem 24 extraída do console da IDE usada na implementação e teste do recomendador, mostra como ficou o cálculo das similaridades:

Figura 24 – Similaridade do Aluno 9 em relação ao restante dos alunos

```
Similaridade com o Aluno 15: 0.11111111111111111
Similaridade com o Aluno 30: 0.4
Similaridade com o Aluno 12: 0.66666666666666666
Similaridade com o Aluno 5: 0.14285714285714285
Similaridade com o Aluno 3: NaN
Similaridade com o Aluno 20: 0.2
Similaridade com o Aluno 27: 0.33333333333333333
```

Se quisermos testar se o cálculo da similaridade seguiu realmente o Coeficiente de Tanimoto, podemos realizar o cálculo de uma das similaridade mostrada na figura 24 manualmente. Calculando a similaridade entre o Aluno 9 e Aluno 30 usando a fórmula do Coeficiente de Tanimoto mostrada na seção 2.3.2, temos $2/5 = 0,4$. Podemos perceber que o resultado é o mesmo mostrado na figura 24 que foi gerado pela execução do módulo.

Agora vamos avançar no próximo passo da execução do módulo de filtragem colaborativa baseado no usuário, vamos para a etapa de definição da vizinhança usando *Threshold*. No nosso teste, definimos um limite inferior como sendo 0.7, portanto, utilizaremos apenas os com similaridade acima de 0.7, sabendo que esse valor varia entre 0 e 1. O aluno mais similar ao Aluno 9, segundo o mostrado na figura 24 é o aluno Aluno 12 (similaridade=0.66). Considerando o limite definido no *Threshold* para obter a vizinhança, não vai ser possível gerar nenhuma recomendação para o aluno 9, pois nenhum outro aluno teve similaridade igual ou acima de 0.7 com o mesmo.

5.1.2.2 Cenário 2 - Recomendação gerada normalmente

Temos a entrada mostrada na figura 25 com questões acertadas, erradas e não respondidas, que servirá de entrada para o módulo de eficiência. O aluno alvo da recomendação será o aluno 9.

Figura 25 – Entrada do módulo de eficiência

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
Aluno 9												
Aluno 2												
Aluno 25												
Aluno 33												
Aluno 17												
Aluno 10												
Aluno 7												
Aluno 29												
Aluno 43												

Sabendo que no cálculo da eficiência os valores variam entre 0 e 1, eliminaremos quem ficar com valor abaixo de 0.7 ao aplicar a fórmula mostrada na seção 4.1 3 (c). Para esse caso teremos dois alunos que serão eliminados, que são o alunos 10 e 43, ambos com eficiência 0.5 e 0.66.

Seguindo agora para o próximo módulo do recomendador, o módulo de filtragem colaborativa baseada no usuário, temos a seguinte entrada após o processamento na entrada figura 25:

Figura 26 – Entrada do módulo de filtragem colaborativa baseado no usuário

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
Aluno 9												
Aluno 2												
Aluno 25												
Aluno 33												
Aluno 17												
Aluno 7												
Aluno 29												

O primeiro passo é realizar o cálculo do Coeficiente de Tanimoto para saber a similaridade entre o Aluno 9 e o restante dos alunos. A imagem 27 extraída do console da IDE usada na implementação e teste do recomendador, mostra como ficou o cálculo das similaridades para esse caso:

Figura 27 – Similaridade do Aluno 9 em relação ao restante dos alunos

```
Similaridade com o Aluno 2: 0.25
Similaridade com o Aluno 25: NaN
Similaridade com o Aluno 33: 0.4
Similaridade com o Aluno 17: 0.8
Similaridade com o Aluno 7: 0.14285714285714285
Similaridade com o Aluno 29: 0.14285714285714285
```

Após o cálculo das similaridades devemos pegar a vizinhança com base no Threshold. Para esse caso, apenas o aluno 17 será usado, pois teve um valor de similaridade de 0.8 que é maior que o Threshold de 0.7.

Já que para esse caso temos recomendações para serem dadas, então vamos remover do resultado parcial as questões que o aluno 17 tem em comum com o aluno alvo, afinal queremos recomendar questões diferentes das que ele já fez. Isso é mostrado na imagem 28 abaixo:

Figura 28 – Resultado parcial após a aplicação da vizinhança

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12
Aluno 9												
Aluno 17												

Analisando a figura 28 percebemos que só pode ser recomendada a questão Q12. Nesse ponto seria feita uma contagem para saber quais questões se repetem mais entre os usuários similares ao alvo e que podem ser recomendadas, as que se repetissem mais teriam uma maior prioridade na hora da recomendação, mas como nesse caso só existe uma questão a ser recomendada, isso não é problema.

Na figura 29 temos o resultado final da recomendação que é disponibilizado pela API em um arquivo JSON e pode ser consumido por qualquer aplicação. Para esse caso apenas a questão 12 foi recomendada. Caso existissem outras questões, elas também entrariam na lista de questões a serem recomendadas, seguindo o critério de ordem por prioridade, recomendando primeiro as questões dos alunos que foram mais similares.

Figura 29 – Recomendações geradas pela API

```
{
  "recomendacoes": {
    "itens": 12
  }
}
```

Tal comportamento também era esperado ao final da execução. Portanto, podemos concluir que o recomendador se comportou como o esperado e apresentou os resultados esperados. Além disso, pelo fato de estar sendo usado métodos amplamente abordados e testados na literatura, então, foi considerado que o recomendador criado tem validade, mesmo sem a possibilidade de predição de notas por causa da sua natureza booleana e aplicação em uma disciplina em andamento.

Um outro teste possível seria para o caso do usuário ser novo no sistema. Para esse cenário a recomendação gerada é a das questões mais resolvidas e acertadas para o

nível do aluno, ou seja, as questões mais populares, e com isso resolver o problema do cold-start inerente a esse tipo de recomendação.

5.2 Diferencial e novidades do presente trabalho

Como diferencial e novidades do presente trabalho, podemos mencionar que na literatura não foi encontrado nenhum tipo de abordagem para geração de recomendações seguindo os critérios definidos nesse trabalho, que passa pelos conceitos da RBT e a recomendação híbrida flexível e facilmente extensível.

Além disso, temos a adoção do estilo de arquitetura REST, que pelo que foi visto na literatura, não é um ponto que tem sido amplamente explorado quando estamos falando de AVAs. Esse estilo de arquitetura pode inspirar a modelagem e construção de vários elementos para AVAs de forma genérica, independente de qual AVA será usado, e isso pode ir muito além do domínio de recomendação.

6 CONCLUSÃO E CONTINUAÇÃO DO TRABALHO

Alinhados com a atual tendência dos Sistemas de Aprendizagem Adaptativos, a presente dissertação apresentou, um sistema de recomendação de questões avaliações com três etapas de refinamento, sendo a primeira baseada em RBT, que classifica o aluno e as questões em um dos níveis do RBT, a segunda levou em consideração a eficiência dos alunos e a terceira baseada em filtragem colaborativa baseada no usuário. A utilização do modelo proposto pode trazer benefícios para estudantes que teriam atividades sugeridas de acordo com seu perfil e nível cognitivo.

6.1 Contribuições

Tendo em vista que no início da pesquisa não tínhamos nenhuma percepção a respeito de como criar uma relação entre alunos e questões nas disciplinas do cursos LC-EAD ofertados no AVA Moodle visando melhorar a evolução do aluno, então, podemos considerar como uma a principal contribuição desta pesquisa, o processo de geração da recomendação com base no modelo proposto, que funde a RBT, um critério de eficiência e filtragem colaborativa baseada no usuário, visando encontrar questões que possibilitem uma melhora na evolução da aprendizagem do aluno.

O presente trabalho se diferencia por ter um modelo que mistura elementos na geração da recomendação que não foram encontrados na literatura, além de ser mais genérico no que diz respeito a como essas informações podem ser usadas. Esse serviço pode ser usado na construção de um plugin do Moodle para passar as recomendações para dentro do ambiente educacional, mas se esse modelo tivesse sido construído dentro do próprio plugin do Moodle por exemplo, ele estaria limitado apenas aquele contexto e com um alto nível de acoplamento, e isso traz prejuízos na evolução e manutenção de qualquer software e de certa forma não seguiria o princípio da coesão.

6.2 Desafios e limitações

O primeiro desafio desta pesquisa foi a obtenção de dados reais para servir de base para nossas análises. Felizmente, como mencionado em outras partes do trabalho, as bases de dados de duas disciplinas do curso LC-EAD foram disponibilizados para esta pesquisa.

Em seguida, já de posse de dados para análise, o outro desafio encontrado, ainda no trabalho de [Carneiro 2018] foi: Como avaliar o aluno de acordo com os objetivos educacionais da RBT sem haver um ambiente pré-formatado segundo essa taxonomia? O fato das questões não terem sido elaboradas de acordo com a RBT dificultou essa análise, pois diversas das questões de avaliações que foram respondidas corretamente pelos alunos, não correspondiam a nenhum nível RBT.

Após o mencionado, entrando agora no que é atribuição deste trabalho, um dos maiores desafios durante a realização da pesquisa foi a investigação de conceitos que pudessem ser aplicados na proposta de um modelo de recomendador adequado aos dados que temos, pois, existem muitos conceitos aplicados na área de recomendação de conteúdo, então identificar e filtrar quais que se encaixam ao propósito foi algo complicado.

Compreender mais detalhes da base de dados do Moodle visando identificar mais dados necessários para a evolução do modelo, além dos que foram obtidos no modelo RBT proposto em [Carneiro 2018] também foi um outro desafio, devido a quantidade de relacionamentos e complexidade da base de dados, por isso não foi possível obter dados temporais, apenas se acrescentou dados relativos as questões respondidas que não foram acertadas. Os dados temporais seriam úteis para saber em que momento determinado aluno alcançou um nível RBT e quando saiu. Esses dados auxiliariam a mensurar se a recomendação melhorou o desempenho, pois sabemos que se trata de alunos similares e isto possibilitaria saber se diferentes alunos alvo conseguiriam ser melhores, piores ou se manteriam a mesma média de tempo para mudança de nível que seus similares.

Uma outra questão foi a integração da base de dados com a aplicação, o Moodle não pode ter seu banco acessado por aplicações para gerenciar banco de dados, mesmo com a aplicação estando autenticada para acessar. Para ter o acesso, é preciso primeiro importar o curso na plataforma para ter acesso a base de dados, e tudo isso se deve ao formato que o curso é disponibilizado, ou seja, é um alto acoplamento, apenas sendo permitido acessar a base de dados após a importação no Moodle, mesmo sabendo que a base de dados não precisaria dessa dependência para funcionar. Após importar o curso, acontecem problemas nas configurações do servidor em relação aos tamanhos dos arquivos e outros detalhes, o que complicaria absurdamente a continuidade do projeto caso fosse necessário configurar tudo novamente. O acesso ao banco de dados previamente foi perdido por fatores técnicos, então seria necessário instalar tudo novamente e passar por todos esses problemas. Porém, já haviam instâncias de dados extraídas em momentos onde nós ainda tínhamos acesso a base de dados, então enfrentar toda essa complexidade para ter a base de dados disponível para integrar ao recomendador poderia comprometer outras partes que precisavam ser feitas. Então o envio dos dados para a aplicação é um ponto que precisa ser melhorado, fornecendo a aplicação acesso direto a base de dados.

A etapa mais interessante e empolgante do trabalho foi ver o recomendador funcionando e perceber o quão rápido ele consegue gerar as recomendações, mesmo para entradas razoavelmente grandes.

Mesmo com esses desafios e limitações, de acordo com os estudos realizados e com os resultados obtidos, o módulo de recomendação usando RBT, módulo de eficiência e filtragem colaborativa proposto demonstrou-se satisfatório e pode ser evoluído levando em conta os pontos citados.

6.3 Trabalhos futuros

O primeiro ponto para o futuro, seria trazer melhoras com relações as limitações faladas na seção 6.2. Além disso, seria interessante procurar, entender e trazer outros elementos que possam não terem sido identificados previamente e que visem melhorar a recomendação. Outro ponto interessante seria criar um plugin no Moodle para consumir as informações geradas pela API e tornar disponível para os alunos dentro do próprio AVA.

REFERÊNCIAS

- [1] PRIMO, T. ; VICCARI, R. M. ; SILVA, J. M. C. . Rumo ao Uso de Metadados Educacionais em Sistemas de Recomendação. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO-SBIE, vol. 1, no. 1, 2010.
- [2] CAZELLA, Sílvio César ; REATEGUI, Eliseo ; MACHADO, Munique ; BARBOSA, José Luis . Recomendação de Objetos de Aprendizagem Empregando Filtragem Colaborativa e Competências. In: Simpósio Brasileiro de Informática na Educação, 2009, Florianópolis. Anais do SBIE, 2009.
- [3] Ferro, M.R.C. . **Modelo de Sistema de Recomendação de Materiais Didáticos para Ambientes Virtuais de Aprendizagem**. Maceió, 2010. Disponível em: <http://www.repositorio.ufal.br/bitstream/riufal/836/1/Dissertacao_%20Marcio%20Roberio%20da%20Costa%20Ferro_%202010.pdf>. Acesso em: 15 mar. 2018.
- [4] CAZELLA, Sílvio; CHAGAS, Irismar; BARBOSA, Jorge; REATEGUI, Eli-seo. Um modelo para recomendação de artigos acadêmicos baseado em filtragem colaborativa aplicado à ambientes móveis. Porto Alegre: UFRGS - Universidade Federal do Rio Grande do Sul, 2008. Disponível em: <<http://www.lume.ufrgs.br/bitstream/handle/10183/22900/000684572.pdf?sequence=1>>. Acesso em: 25 jul. 2018.
- [5] ADOMAVICIUS, G.; TUZHILIN, A. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. **IEEE Transactions on Knowledge and Data Engineering**, p. 734-749, jun. 2005. Disponível em: <http://blog.ag-nbi.de/wp-content/uploads/2015/10/adomavicius-recsys.pdf>. Acesso em: 28 jul. 2018.
- [6] Drachsler, K, H.; Verbert, O, C.; Santos, N. (2015) “Panorama dos sistemas de recomendação para apoiar a aprendizagem”. Manual sobre sistemas de recomendação.
- [7] HERLOCKER, J. L. **Understanding and Improving Automated Collaborative Filtering Systems**. 2000. 144p. Tese. University of Minnesota, Minnesota, EUA.
- [8] Bezerra, B. L. D.. Uma Solução em filtragem de informação para sistemas de recomendação baseada em análise de dados simbólicos. Recife: UFPE - Universidade Federal de Pernambuco, 2004. Disponível em: <https://repositorio.ufpe.br/handle/123456789/2475>. Acesso em: 27 set. 2019.
- [9] OWEN, S. et al. **Mahout in Action**. Shelter Island: Manning Publication Co., 2012.

- [10] MOORE, Jonathan; CHURCHWARD, Michael. **Moodle 1.9 Extension Development**. Birmingham: Packt Publishing, 2010.
- [11] JANNACH, D; ZANKER, M; FELFERNIG, A; FRIEDRICH, G. **Recommender Systems: An Introduction**. New York: Cambridge, 2011.
- [12] Carneiro, J. N. **Aplicação de Learning Analytics para Modelagem do Aluno de acordo com a Taxonomia de Bloom Revisada**. Dissertação – Programa de Pós-Graduação em Informática, Universidade Federal da Paraíba, João Pessoa, 2018.
- [13] Martins, D; Tiziotto, S; Cazarini, E. Ambientes Virtuais de Aprendizagem (AVAs) como ferramentas de apoio em Ambientes Complexos de Aprendizagem (ACAs). **Revista Brasileira de Aprendizem Aberta e a Distância**, v.15, p. 113-128, 2016.
- [14] Hassan, M. and Hamada, M. (2017). Smart media-based context-aware recommender systems for learning: A conceptual framework. In 2017 16th International Conference on Information Technology Based Higher Education and Training (ITHET), pages 1–4. IEEE.
- [15] O ambiente Virtual de Aprendizagem. Ituverava: NEAD - Núcleo de Educação a Distância. Disponível em: <<http://www.nead.feituverava.com.br/index.php/faq/30-o-ambiente-virtual-de-aprendizagem>>. Acesso em: 5 de out. de 2018
- [16] O ambiente Virtual de Aprendizagem. Ituverava: NEAD - Núcleo de Educação a Distância. Disponível em: <<http://www.nead.feituverava.com.br/index.php/faq/27-quais-as-vantagens-da-educacao-a-distancia>>. Acesso em: 5 de out. de 2018
- [17] Silva, C; Figueiredo, V. Ambiente virtual de aprendizagem: comunicação, interação e afetividade na EAD. **Revista Aprendizagem em EAD**, v. 1, p. 1-14, Outubro, 2012.
- [18] Almeida, M. Educação a distância na internet: abordagens e contribuições dos ambientes digitais de aprendizagem. **Educação e Pesquisa**, v. 29, n. 2, p. 327-340, 2003.
- [19] Gabardo, P; Quevedo, S; Ulbricht, V. ESTUDO COMPARATIVO DAS PLATAFORMAS DE ENSINO-APRENDIZAGEM. **Revista Brasileira de Aprendizem Aberta e a Distância**, 2010.
- [20] Gabardo, P; Quevedo, S; Ulbricht, V. Estudo comparativo das plataformas de ensino-aprendizagem. **Revista Eletrônica de Biblioteconomia e Ciência da Informação**, v. 15, n. Esp., p. 65-84, 2010.

- [21] Costa, E; Aguiar, J; Magalhães, J. Sistemas de Recomendação de Recursos Educacionais: conceitos, técnicas e aplicações. In: Congresso Brasileiro de Informática na Educação, II, São Paulo-SP, 25 a 29 de novembro de 2013.
- [22] Lázaro, A. S. **ANÁLISE E SELEÇÃO DE ALGORITMOS DE FILTRAGEM DE INFORMAÇÃO PARA SOLUÇÃO DO PROBLEMA COLD-START ITEM**. Dissertação – Departamento de Ciência da Computação, Universidade Federal de Lavras, Lavras, 2010.
- [23] Silva, R. G. N. **Sistema de Recomendação baseado em conteúdo textual: avaliação e comparação**. Universidade Estadual de Feira de Santana (UEFS), Feira de Santana, 2014.
- [24] Moodle (2013). Moodle Plugins Directory: Recommender. Disponível em: https://moodle.org/plugins/view.php?plugin=block_recommender. Acesso em: 14 dez. 2019.
- [25] [Anderson e Krathwohl 2001]ANDERSON, L.; KRATHWOHL, D. Taxonomy for learning, teaching and assessing: A revision of Bloom’s Taxonomy of educational objectives. [S.l.:s.n.], 2001.
- [26] [Costa et al. 2013]COSTA, R. da et al. Ferramentas avaliativas disponíveis em um ambiente virtual de aprendizagem usada no planejamento de um curso através do mapa de dependências. In: Novas Tecnologias em Educação. [s.n.], 2013. Disponível em: <http://seer.ufrgs.br/renote/article/viewFile/41633/26412>.
- [27] [Peckham e McCalla 2012]PECKHAM, T.; MCCALLA, G. Mining student behavior patterns in reading comprehension tasks. In: EDM. [S.l.: s.n.], 2012.
- [28] [Ferraz e Belhot 2010]FERRAZ, A. P. do C. M.; BELHOT, R. V. Taxonomia de bloom: revisão teórica e apresentação das adequações do instrumento para definição de objetivos instrucionais. Gestão & Produção, FapUNIFESP (SciELO), v. 17, n. 2, p. 421–431, 2010.
- [29] [Santos 2016]SANTOS, R. dos. **INSERINDO A TAXONOMIA REVISADA DE BLOOM EM UM MOOC**. mathesis, 2016. Disponível em: <https://ppgcc.ufersa.edu.br/wpcontent/uploads/sites/42/2014/09/INSERINDO-A-TAXONOMIA-REVISADA-DEBLOOM-EM.pdf>.
- [30] [Souto 2003]SOUTO, M. Diagnóstico on-line do Estilo Cognitivo de Aprendizagem do Aluno em um Ambiente Adaptativo de Ensino e Aprendizagem na Web: uma abordagem empírica baseada na sua trajetória de aprendizagem. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul, Porto Alegre, 2003. Disponível em: <http://www.lume.ufrgs.br/bitstream/handle/10183/4837/000416006.pdf?sequence=1>.

- [31] F. Ricci, B. Shapira, and L. Rokach, *Recommender systems handbook*, Second edition, 2015.
- [32] Dezebembro, G. D. **Uma medida de similaridade híbrida para a correspondência aproximada de múltiplos padrões**. Universidade de São Paulo (USP), São Paulo, 2019.

ANEXO A – ANEXOS E APÊNDICES 1