

Sistema de Recomendação Híbrido Utilizando as Técnicas de Filtragem Colaborativa e Baseada em Conteúdo

Lucas Magnus da Silva¹, Merisandra Côrtes de Mattos Garcia¹

¹Curso de Ciência da Computação – Grupo de Pesquisa em Inteligência Artificial Aplicada – Universidade do Extremo Sul Catarinense (UNESC)
Criciúma – SC – Brasil

lucasmagnusdasilva@gmail.com, mem@unesc.net

Abstract. Recommendation systems combine techniques such as unsupervised machine learning to recommend personalized items for users based on their choices. This research combines collaborative filtration and based on content technics, developing a hybrid recommendation system for movies through the characteristics addition method and use of contextual pre-filtering. After 42 experiments carried out in groups of 50 users up to 350 with data from the MovieLens database, the evaluation methods used showed variation values less than 1, indicating good results, but the data set has proved limited for the application of contextual pre-filtration and based on content filtration.

Resumo. Sistemas de recomendação combinam técnicas como aprendizado de máquina para recomendar itens personalizados a usuários com base em suas preferências. Esta pesquisa combina as técnicas de filtragem colaborativa e baseada em conteúdo, desenvolvendo um sistema de recomendação híbrido para filmes por meio do método de acréscimo de características e utilização da pré-filtragem contextual. Após 42 experimentos realizados em grupos de 50 usuários até 350 com os dados da base MovieLens, os métodos avaliativos utilizados apresentaram variação menor que 1, indicando bons resultados, porém o conjunto de dados experimentais se mostrou limitado a aplicação da pré-filtragem contextual e da filtragem baseada em conteúdo.

1. Introdução

Ao longo dos anos, com o aumento na disponibilidade de dados na internet, técnicas para auxiliar o usuário a encontrar conteúdos que sejam de seu interesse tornaram-se indispensáveis, sendo uma dessas técnicas os sistemas de recomendação [Seyednezhad et al. 2018].

Sistemas de recomendação são ferramentas utilizadas para solucionar o problema de sobrecarga de dados, exibindo para o usuário o conteúdo que mais se assemelha aos seus interesses. As recomendações são calculadas por meio de métodos que buscam associar o gosto do usuário ao conteúdo disponível, formando um perfil que resume o seu comportamento [Irfan et al. 2019]. A utilização de sistemas de recomendação pode ser exemplificada como uma tecnologia analítica que calcula a probabilidade de um produto ser escolhido por uma determinada pessoa [Park et al. 2010]. Este modelo de sistema utiliza algoritmos de aprendizado de máquina não supervisionado, ramo da inteligência artificial, caracterizado por identificar padrões a partir de dados.

Sistemas de recomendação são importantes em diversos campos e são amplamente utilizados em vários domínios, como em compras (Amazon), na sugestão de músicas (Pandora), de filmes (Netflix), viagens (Trip-Advisor), no âmbito alimentício (Yelp), na indicação de perfis sociais (Facebook) e artigos (Ted) [Al-Ghuribi e Mohd Noah 2019].

Sistemas de recomendação obtêm seus dados de forma explícita ou implícita. Na forma explícita o usuário indica quais são as suas preferências de forma espontânea, por meio da atribuição de uma nota de valor numérico ou de uma avaliação positiva ou negativa. Na forma implícita são considerados fatores subentendidos de interesse como número e duração do acesso e históricos de uso [Reategui 2005 apud Barbosa 2014].

Diferentes técnicas são utilizadas, sendo as mais comuns a filtragem colaborativa e a filtragem baseada em conteúdo [Gatzioura et al. 2019]. A baseada em conteúdo utiliza o histórico do usuário buscando recomendar itens semelhantes aos que já foram avaliados por ele, por exemplo, ao avaliar um filme com boa nota, outros do mesmo gênero e atores podem ser recomendados. Assim, não precisa de uma grande quantidade de usuários envolvidos, pois a recomendação é realizada por meio da similaridade entre os itens avaliados pelo próprio usuário e itens novos que ainda não avaliou [Peireira 2016].

A filtragem colaborativa identifica usuários com preferências similares ao usuário fornecido e realiza as recomendações, podendo trazer mais confiança no item recomendado [Seyednezhad et al. 2018; Shardanand and Maes 1995]. Essa técnica de filtragem busca a similaridade entre os usuários que utilizam o sistema de recomendação. Ainda que seja possível utilizar os itens como parâmetro, nesse modelo o conteúdo do item não é o mais relevante, mas sim quem já o avaliou [Shaikh et al. 2017]. A filtragem colaborativa baseada em usuário busca pessoas com perfis semelhantes e gera recomendações, a partir de avaliações desses usuários com similares. Na identificação de dois perfis semelhantes são utilizados algoritmos de similaridade [Peireira, 2016].

A filtragem colaborativa apresenta problema quando a matriz de avaliação do usuário é muito esparsa, ou seja, não existe um padrão parecido de interesse, tendo-se uma semelhança muito pequena com outros usuários [Yang et al. 2016]. Na filtragem baseada em conteúdo tem-se o problema que é conhecido como superespecialização, pois o usuário fica limitado a itens similares aos que já teve interesse, sendo assim, itens novos podem não ser recomendados [Lázaro 2010]. Visando contornar estes problemas e melhorar os sistemas de recomendação que fazem uso dessas técnicas, tem-se a filtragem híbrida que reúne diferentes técnicas de recomendação em busca de combinar vantagens e atenuar desvantagens [Kim et al. 2006; Yadav 2018]. Este tipo de filtragem busca aumentar o desempenho dos resultados obtidos pelo sistema ao combinar as melhores características de duas ou mais técnicas de recomendação [Chen et al. 2018; Yamaguti e Oliveira 2013].

A filtragem colaborativa, de acordo com Gatzioura et al. (2019) e Schafer et al. (2007), é a técnica de recomendação mais utilizada devido a sua eficácia, baseando-se na troca de experiência entre os usuários que possuem interesses similares, sendo assim é possível que sejam recomendados itens diversos, essa variação nos itens recomendados é importante neste tipo de sistema [Barbosa 2014]. A filtragem baseada em conteúdo, por não depender das recomendações de outros usuários, tende a sofrer menos quando a base de dados do sistema tem poucos dados, problema típico da filtragem colaborativa que ocorre quando o usuário recém começou a interagir com o sistema e existem poucos dados sobre ele, fazendo com que o sistema de recomendação não tenha informações suficientes

para avaliar [Adomavicius et al. 2011; Irfan et al. 2019]. Assim, o hibridismo entre essas técnicas pode proporcionar melhor performance e atenuar os efeitos negativos de cada uma delas se utilizadas separadamente [Burke 2002; Gatzoura et al. 2019].

Na presente pesquisa, a partir do seu objetivo geral, desenvolveu-se um sistema de recomendação híbrido para filmes, combinando-se as técnicas de filtragem baseada em conteúdo e colaborativa, com a finalidade de se obter melhores resultados de recomendações para os usuários, atenuando a desvantagem da superespecialização e ampliando o domínio de itens recomendados. Tendo-se também como objetivos: descrever as técnicas de filtragem baseada em conteúdo e colaborativa; aplicar o hibridismo por meio das técnicas de filtragem baseada em conteúdo e colaborativa; empregar o algoritmo K-Nearest Neighbors (KNN); aplicar a pré-filtragem contextual; analisar os resultados pelos métodos avaliativos probabilísticos de Erro Médio Absoluto, Erro Médio Quadrático e Raiz Quadrada do Erro Médio.

Um dos domínios de aplicação amplamente utilizado é a recomendação de filmes. Os sistemas de recomendação de filmes ajudam os usuários a encontrarem conteúdos de sua preferência a partir de uma biblioteca com milhares de itens. Além disso, as pesquisas na área costumam empregar conjuntos de dados baseados em filmes para confirmarem os resultados dos seus algoritmos de recomendação [Yin et al. 2014].

A filtragem colaborativa nesta pesquisa foi desenvolvida por meio do algoritmo de aprendizado de máquina não supervisionado dos k vizinhos mais próximos, do inglês K - Nearest Neighbors (KNN), que segundo Bobadilla et al (2011) busca determinar a vizinhança k do usuário com base em suas classificações e assim prever as avaliações do usuário fornecido.

2. Trabalhos relacionados

Na literatura, conforme apresentado no Quadro 1, existem internacionais e nacionais relacionados a sistemas de recomendação, dentre estes, pode-se citar a pesquisa de Kristoffersen, Shepstone e Tan (2020) que demonstra a importância da utilização do contexto para a geração de recomendações. Os dados utilizados foram obtidos a partir de um estudo realizado com cem pessoas durante um período de cinco semanas, relacionando um conjunto de dados televisivos com informações contextuais. A forma de geração das recomendações foi a partir da filtragem baseada em conteúdo utilizando o contexto. A pesquisa apresenta que determinados gêneros são preferidos durante os dias de semana e outros em finais de semana, dessa forma observaram que o período da semana influencia na avaliação dos usuários. Os resultados experimentais mostraram que a inclusão de informações contextuais sociais e temporais melhoram a precisão e a diversidade das recomendações.

Do, Le e Yoon (2020) desenvolveram um sistema de recomendação híbrido, utilizando filtragem colaborativa e baseada em conteúdo com utilização da estratégia de ponderação dinâmica. O conjunto de dados MovieLens foi utilizado como entrada para as recomendações. A aplicação do hibridismo foi realizada a partir dos resultados obtidos pelas técnicas de recomendação utilizadas e empregou-se um peso para cada recomendação por meio de métricas que atribuíam mais confiança à uma recomendação. A filtragem híbrida proposta obteve melhores resultados que a colaborativa baseada em itens quando analisado com poucos itens, mas para um número maior de itens ambas as técnicas convergiram devido a colaborativa tender a recomendar os itens mais populares.

Fessahaye et al. (2019) propõem um sistema de recomendação híbrido para recomendação de músicas utilizando a filtragem colaborativa e a baseada em conteúdo como entrada para um modelo de classificação de aprendizado de máquina. Essa entrada é alimentada em uma rede neural profunda e, por meio de um processo de treinamento, a rede aprende a reconhecer padrões no histórico de escuta de um usuário, recomendando músicas que tem a confiança de que o usuário apreciará. Os resultados obtidos mostraram que o sistema proposto obteve até 88% de precisão em determinados casos.

Bezerra et al. (2018) apresentam um sistema de recomendação híbrido integrado a uma ontologia que gerencia as informações de perfil do aprendiz e de conteúdos provenientes do Youtube e Wikipédia. Os resultados mostraram que as recomendações geradas estavam de acordo com as necessidades dos estudantes e coerentes com as disciplinas do seu curso, como também de seus interesses. A respeito da adaptação do sistema conforme a utilização pelo usuário relataram que as interações realizadas modificaram os conteúdos que foram recomendados, conseguindo identificar a mudança de interesse dos estudantes e assim sugerir itens semelhantes aos que eles interagiram.

Avancini (2016) propôs um modelo de sistema de recomendação utilizando a filtragem colaborativa e adicionando informações contextuais. Dessa forma, visava gerar recomendações mais precisas de atividades de lazer a um usuário baseado em seus interesses e contexto. A identificação dos interesses dos usuários se deu por meio de um aplicativo desenvolvido. Os resultados mostraram que o contexto utilizado foi demasiadamente simples, pois só empregou o dado do clima. A recomendação de atividades de lazer pelo sistema de recomendação proposto utilizando o contexto do usuário foi realizada, porém os resultados obtidos se mostraram limitados.

Trabalho	Dados	Estratégia	Hibridismo	Recomendação	Resultados	Local
Kristoffersen Shepstone e Tan (2020)	Estudo com cem usuários	Baseada em conteúdo e contexto	-	Programa de TV	Melhorias nas recomendações com contexto temporal e social	Aalborg, Dinamarca
Do et al. (2020)	MovieLens	Híbrido de colaborativa e baseada em conteúdo com pesos dinâmicos	Método Balanceado/ Baseada em usuário e item	Filme	Abordagem híbrida teve melhores resultados para poucos itens	Hanoi, Vietnã
Fessahaye et al. (2019)	Spotify Dataset	Híbrido colaborativa e baseada em conteúdo com <i>machine learning</i>	Método em Níveis/Baseada em item	Música	Melhoria de até 88% de precisão de recomendações	Las Vegas, Estados Unidos
Bezerra et al. (2018)	Youtube e Wikipédia	Híbrido colaborativa e baseada em conteúdo com ontologia	Método em Níveis/Baseada em item	Conteúdo para educação	A super-especialização influenciou negativamente	Mossoró, Brasil
Avancini (2016)b	Aplicativo	Colaborativa com contexto	-	Atividade de lazer	A limitação do contexto prejudicou os resultados	Florianópolis, Brasil
Esta pesquisa	MovieLens	Híbrido colaborativa e baseada em conteúdo com e sem Pré-filtragem contextual	Método de Acréscimo de Características/ Baseada em usuário e item	Filme	-	Criciúma, Brasil

Quadro 1. Comparativo entre os trabalhos relacionados.

O Quadro 1 apresenta um comparativo entre os trabalhos relacionados, informando os dados que foram utilizados em cada trabalho, a estratégia de recomendação, o que foi recomendado e o local onde foram realizados.

3. Materiais e métodos

O desenvolvimento do sistema de recomendação híbrido baseado em contexto proposto por esta pesquisa consiste em uma pesquisa aplicada de base tecnológica, desenvolvendo uma aplicação prática, descritiva, realizando análise dos resultados obtidos, e bibliográfica, sendo realizadas pesquisas em artigos e teses [Wazlawick 2014]. A finalidade desta pesquisa é a utilização de duas técnicas de recomendação, sendo estas a filtragem baseada em conteúdo e a filtragem colaborativa, aplicando o hibridismo entre elas. Buscando filtrar apenas os dados que se enquadrem no contexto do usuário, foi utilizada a técnica de pré-filtragem contextual como entrada de dados. A Figura 1 apresenta a arquitetura geral do sistema de recomendação proposto, acessível a partir de um aplicativo móvel, realizando a conexão com o servidor, responsável pelo algoritmo de recomendação acessando os dados do banco de dados, inicializado com os dados do conjunto MovieLens, e aplicada a pré-filtragem contextual. Posteriormente, com estes dados é realizada a filtragem colaborativa e depois a baseada em conteúdo formando o hibridismo por meio do método de acréscimo de características, para então realizar a recomendação e enviar por uma *Application Programming Interface* (API) para o aplicativo.

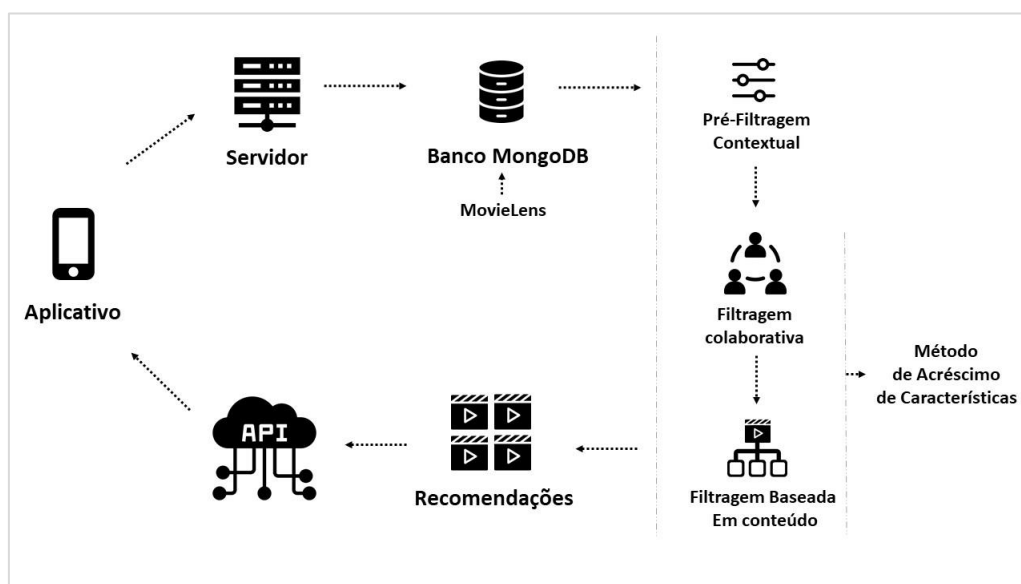


Figura 1. Arquitetura do projeto

3.1 Conjunto de dados

O conjunto de dados utilizado foi o MovieLens *small* destinado a educação e desenvolvimento, constituído de 100.836 avaliações, 9,742 filmes, 600 usuários e 20 gêneros, atualizados até setembro de 2018 pela organização responsável GroupLens Research. A disposição das notas neste conjunto de dados [Gráfico 1] tem como as mais atribuídas pelos usuários as avaliações 3 e 4. Essa base é amplamente utilizada para projetos educacionais e testes de sistemas de recomendação. A base MovieLens é gratuita, tem cerca de 1 *megabyte* de tamanho e pode ser encontrada no site da própria GroupLens¹.

¹ Disponível em <https://grouplens.org/datasets/movielens/>

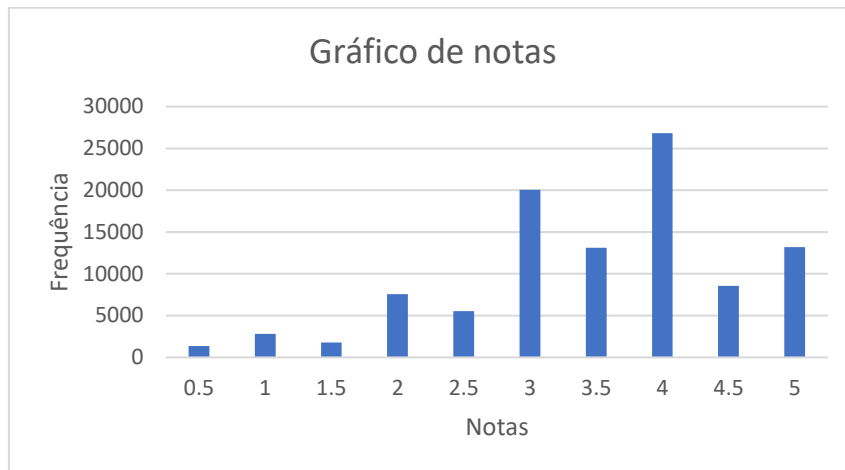


Gráfico 1. Notas da base MovieLens

3.2 Servidor

O servidor da aplicação foi desenvolvido na linguagem Python versão 3.9, utilizando como ambiente de programação a IDE PyCharm Community Edition 2020.2.3 no sistema operacional Windows 10. A arquitetura do servidor foi projetada para realizar as recomendações a partir do servidor e enviar para a aplicação móvel por meio do micro-framework Flask versão 1.1, criando uma API de comunicação entre o servidor e o aplicativo móvel.

O banco de dados desta pesquisa foi desenvolvido e estruturado [Figura 2] no MongoDB na sua versão 4.4.1 e controlada pela interface MongoDB Compass versão 1.23. A escolha pelo MongoDB foi devido a sua estrutura *NoSQL*, permitindo uma melhor distribuição e mais rapidez na organização e análise de grande quantidade de dados, sendo recomendado para o projeto desenvolvido [Sharma and Kaur 2019]. Outro fator importante para a escolha desse banco de dados é o fato de ser um *software* de código aberto. A integração entre a IDE PyCharm e o banco de dados MongoDB foi realizada por meio da biblioteca *pymongo* versão 3.11, disponibilizada pelo gerenciador de pacotes de bibliotecas *Python Package Index* (PIP) na versão 8.0.2.

links	movies	users	ratings	mylist
<pre>{ _id: <ObjectId> movied: <String> imdbId: <String> tmdbId: <String> }</pre>	<pre>{ _id: <ObjectId> movied: <String> title: <String> genres: <Array> year: <String> }</pre>	<pre>{ _id: <ObjectId> userId: <String> name: <String> phone: <String> email: <String> createdAt: <String> }</pre>	<pre>{ _id: <ObjectId> userId: <String> movied: <String> rating: <String> timestamp: <String> }</pre>	<pre>{ _id: <ObjectId> userId: <String> movied: <String> tmdbId: <String> timestamp: <String> }</pre>

Figura 2. Estrutura do banco de dados

3.3 Aplicação móvel

Foi desenvolvida uma aplicação móvel para apresentação dos resultados do algoritmo híbrido na linguagem Dart 2.10.3, utilizando o kit de desenvolvimento de interface Flutter na versão 1.21 e programado na IDE Android Studio, versão 3.6. O framework Flutter permite disponibilizar o aplicativo para as plataformas Android e IOS. No

desenvolvimento da aplicação foram utilizadas as bibliotecas `cupertino_icons 1.0` para o acesso a ícones predefinidos, `http 0.12.2` para realização de requisições do servidor, `flutter_spinkit 4.1.2` para a exibição de ícones de carregamento nas requisições, `shared_preferences 0.5.10` para a criação do banco de dados local e as bibliotecas do Firebase: `firebase_core 0.5.0`, `firebase_analytics 6.0.0` e `firebase_auth 0.18.0` para as funções de autenticação e gerenciamento. Estas bibliotecas foram integradas a partir do gerenciador de pacotes Pub.

A aplicação foi estruturada a partir do padrão de projetos *Model-View-Controller* (MVC), organizando assim o projeto em modelos, controladores e parte visual. A comunicação entre o servidor e a aplicação móvel foi feita por meio de uma API. A API para os filmes recomendados retorna o código do filme na plataforma TMDb, essa no qual contém informações detalhadas sobre cada filme, permitindo exibir a capa do filme, nome, sinopse e elenco. A autenticação dos usuários dentro do aplicativo é feita por meio do Firebase Auth utilizando o modo de autenticação de e-mail e senha, desta forma a senha do usuário fica restrita aos servidores do Firebase, essa ferramenta foi escolhida para a realização da autenticação devido a sua fácil integração e por ser gratuita. A [Figura 3] apresenta as telas principais do aplicativo, composta pela tela de *login* da aplicação [Figura 3a], a tela inicial [Figura 3b] em que apresenta o menu e recomendações geradas pelo algoritmo proposto e a tela de detalhes de um filme [Figura 3c] que apresenta as informações de capa, título, título original, ano de lançamento, gêneros, sinopse e elenco do filme e o *rating bar* para avaliação.

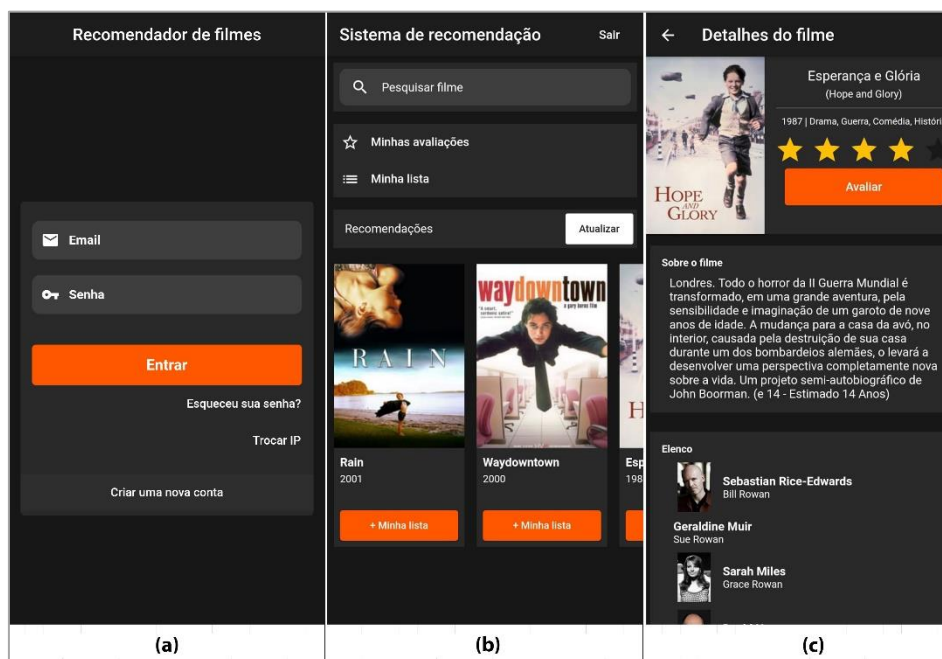


Figura 3. Telas do aplicativo

3.4 Algoritmo de recomendação

Os dados de entrada para o algoritmo são os dados inseridos no banco de dados MongoDB. A hibridização foi realizada por meio do Método de Acréscimo de Características, ela incorpora os resultados obtidos em uma primeira etapa de recomendação a uma segunda etapa, utilizando técnicas diferentes [Primo 2013].

No sistema proposto foi aplicada a técnica de pré-filtragem contextual, limitando os dados de entrada ao contexto do usuário. Essa técnica utiliza informações contextuais para selecionar os dados mais relevantes antes do sistema gerar as recomendações. Uma vantagem dessa abordagem é que ela pode ser implantada em qualquer uma das tradicionais técnicas de sistemas de recomendação. Por exemplo, se uma pessoa quiser ver um filme no domingo, apenas os dados que foram classificados aos fins de semana serão utilizados para gerar a recomendação [Adomavicius et al. 2011]. O processo da pré-filtragem contextual se inicia pela sua entrada de dados, após o contexto é aplicado, antes de gerar a recomendação, para então criar uma lista de itens recomendados junto ao contexto e apresentar ao usuário.

A técnica de pré-filtragem contextual utilizada baseou-se em uma das formas de aplicação proposta na pesquisa de Kristoffersen, Shepstone e Tan [2020] que consiste na aplicação do contexto nos dias da semana. Essa técnica identifica que os gostos dos usuários podem mudar dependendo se ele está no meio ou no final da semana. Para isso utilizou-se o método de obtenção do contexto chamado inobservável, no qual se obtém informações contextuais de forma implícita, no caso o dado *timestamp* da coleção *ratings* do banco de dados, este dado identifica quando o usuário realizou a avaliação. O sistema então identifica se o usuário está no fim de semana ou não, e seleciona apenas as avaliações que se enquadrem neste aspecto para então iniciar o processo de recomendação. Nessa técnica utilizou-se o método manual de generalização do contexto, definindo sábado e domingo como fins de semana e os demais dias como meio da semana.

Após aplicar a pré-filtragem contextual, o sistema entra no processo de gerar a recomendação a partir da filtragem colaborativa. Nesta etapa o sistema busca encontrar os usuários que sejam semelhantes a quem está sendo gerada a recomendação. Para isso é utilizado o algoritmo *K-Nearest Neighbor* (KNN) que é um dos mais utilizados em sistemas de recomendação devido a sua baixa complexibilidade e bom desempenho, sua função é detectar os nós mais próximos com os nós de destino [Fathoni et al. 2018]. Para a identificação de semelhanças, o KNN mede a similaridade de um usuário-alvo com os outros usuários, sendo a função mais conhecida para medir a similaridade e a utilizada neste trabalho assim como proposto por [Santana 2018] o cálculo da distância euclidiana.

A distância euclidiana [Equação 1] é a medida de distância entre dois pontos, fundamentada no teorema de Pitágoras. A distância é calculada com a raiz quadrada da soma das distâncias quadradas. A distância entre dois pontos representa a similaridade entre os usuários, quanto menor a distância, mais semelhantes eles são [Santana 2018].

$$W_{a,u} = \sqrt{\sum_{i=1}^n (x_i - y_{ui})^2} \quad (1)$$

Os dados para o cálculo da similaridade foram as informações da coleção *ratings* no banco dados, permitindo comparar o usuário a quem está sendo gerada a recomendação com todos os outros, formando uma listagem dos usuários e sua similaridade com o usuário alvo, com seu valor variando de 0 a 1. Após definir a similaridade com cada usuário o sistema multiplica a similaridade com a nota que os usuários atribuíram a cada filme, gerando assim uma possível nota que o usuário alvo realizaria. As notas resultantes mais altas são enviadas para o processo seguinte, a aplicação da filtragem colaborativa [Figura 4].

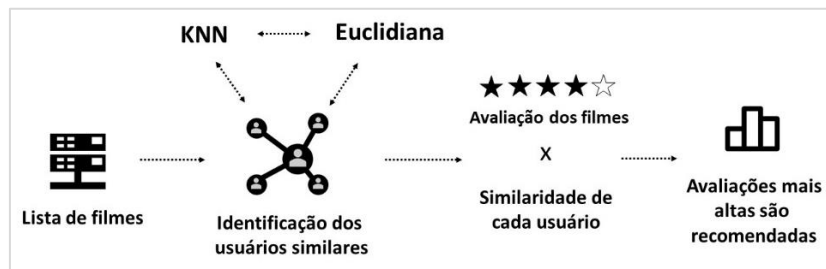


Figura 4. Processo de recomendação da filtragem colaborativa

A partir dos resultados obtidos anteriormente é aplicada a filtragem baseada em conteúdo, realizando assim a hibridização. Este método híbrido é denominado Método de Acréscimo de Características, ele recupera os resultados obtidos na primeira etapa de recomendação e incorpora como entrada para a segunda etapa do algoritmo, neste caso o sistema gera recomendações de filmes com características semelhantes aos recomendados pela filtragem colaborativa. Para a aplicação da filtragem baseada em conteúdo o algoritmo utilizou os gêneros dos filmes registrados no campo *genres* na coleção *movies*, assim como proposto no trabalho de Do, Le e Yoon (2020). A partir destes dados foi possível definir uma semelhança entre os filmes utilizando o Índice de Jaccard [Equação 2], que é uma estatística usada para medir a semelhança entre conjuntos de dados por meio da divisão da intersecção pela união dos dados selecionados. O filme a qual se buscou encontrar outros filmes semelhantes foi comparado com todos os outros e definindo um valor com base no cálculo de Jaccard, indo de 0 para filmes com gêneros completamente diferentes até 1 para filmes com gêneros idênticos. Após definir este valor, foi multiplicada a nota gerada para o filme alvo com a semelhança entre os dois filmes mais similares a ele e recomendados para o usuário.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (2)$$

3.5 Experimentos

Nesta pesquisa foram realizados 42 experimentos [Quadro 2], executando-se o algoritmo híbrido, a filtragem colaborativa e a baseada em conteúdo, em um computador com processador AMD 6300 Black Edition 3.50GHz e 4GB de memória RAM. Os resultados obtidos foram analisados por meio de métricas probabilísticas. Os experimentos foram agrupados em conjuntos de 50 usuários como proposto por Santana (2018), visando a identificação de como os valores se comportam a medida que mais usuários utilizam o sistema de recomendação, chegando até 350 usuários. A base de dados utilizada nos experimentos foi a do MovieLens, empregando-se 5.000 recomendações por meio da filtragem colaborativa e mais uma recomendação a partir da baseada em conteúdo para cada recomendação da colaborativa, podendo-se assim atingir os 10.000 filmes presentes na base MovieLens.

As métricas utilizadas para medir a precisão de cada algoritmo foram a raiz quadrada do erro médio, do inglês Root Mean Square Error (RMSE), o erro quadrático médio, do inglês Mean Squared Error (MSE) e o erro médio absoluto, do inglês Mean Absolute Error (MAE). Estas métricas foram calculadas para cada usuário e realizada uma média dentro do conjunto de usuários do experimento. O melhor resultado possível para as três métricas nestes experimentos é 0, significando assim que a real nota que o usuário atribuiria e a nota que o sistema calculou é a mesma e não varia, já o pior resultado

possível para as três métricas é 5, significando que as notas são completamente diferentes. A escolha para aplicação do MSE deu-se em função da sua característica de distinguir grandes diferenças de variação de forma mais severa (Barbosa, 2014). As métricas RMSE e MAE foram empregadas devido a sua ampla utilização em sistemas de recomendação [Huynh et al. 2020].

Nº	Contexto	Usuários	Dados	Algoritmo	Baseado em	KNN	Avaliação
1	Sim	50, 100, 150, 200, 250, 300, 350	MovieLens	Colaborativo	Usuário	Sim	RMSE, MSE, MAE
2	Sim	50, 100, 150, 200, 250, 300, 350	MovieLens	Baseado em conteúdo	Item	Não	RMSE, MSE, MAE
3	Sim	50, 100, 150, 200, 250, 300, 350	MovieLens	Híbrido pelo método de acréscimo de características	Usuário e item	Sim	RMSE, MSE, MAE
4	Não	50, 100, 150, 200, 250, 300, 350	MovieLens	Colaborativo	Usuário	Sim	RMSE, MSE, MAE
5	Não	50, 100, 150, 200, 250, 300, 350	MovieLens	Baseado em conteúdo	Item	Não	RMSE, MSE, MAE
6	Não	50, 100, 150, 200, 250, 300, 350	MovieLens	Híbrido pelo método de acréscimo de características	Usuário e item	Sim	RMSE, MSE, MAE

Quadro 2. Experimentos realizados

A equação RMSE é definida na Equação (3), tendo-se y como a nota real atribuída pelo usuário, x a nota que o sistema de recomendação calculou, a diferença entre x e y é elevada ao quadrado realizando a soma do resultado até i , que é iniciado em 1, chegue ao valor de n que é o número total de valores comparados, a soma geral é dividida por n , com o resultado obtido é aplicada a raiz quadrada sobre este valor para se obter o RMSE.

$$RSME = \sqrt{\frac{\sum_{i=1}^n (y_i - x_i)^2}{n}} \quad (3)$$

A Equação (4) do MSE se diferencia da equação RMSE por não realizar o cálculo da raiz quadrada ao final da soma.

$$MSE = \frac{\sum_{i=1}^n (y_i - x_i)^2}{n} \quad (4)$$

O MAE, Equação (5), é calculado a partir da soma das diferenças entre as notas atribuídas de verdade pelo usuário com as notas calculadas pelo sistema de recomendação, onde i é a variável inicial, indo até n que representa a quantidade total de notas atribuídas, sendo a diferença calculada em módulo, com o resultado obtido é dividido pelo valor de n , definido assim o valor MAE.

$$MAE = \frac{\sum_{i=1}^n |y_i - x_i|}{n} \quad (5)$$

Os dados coletados foram analisados com auxílio do Software GraphPad Prism versão 5.0. As variáveis foram expressas por meio de média e desvio padrão, enquanto os resultados foram demonstrados por meio de tabelas.

As análises inferenciais foram realizadas com nível de significância $\alpha = 0,05$ e, portanto, confiança de 95%. A homogeneidade das variâncias foi avaliada por meio do teste de Bartlett. Na comparação das médias entre os grupos aplicou-se o teste ANOVA de uma via, seguido de *post hoc* teste de Tukey.

4. Resultados

Os resultados obtidos nesta pesquisa pelo sistema de recomendação híbrido utilizando a pré-filtragem contextual, foram comparados com os resultados obtidos pela filtragem colaborativa e com a baseada em conteúdo sem utilização de contexto, utilizando as métricas RMSE, MSE e MAE. A Tabela 1 mostra a média, o desvio padrão e o valor de significância p para cada uma das métricas utilizadas, sendo agrupados em conjuntos de 50, 100, 150, 200, 250, 300 e 350 usuários. Os valores p maiores que 0,05 indicam que não houve diferença estatisticamente significativa, já os valores menores ou iguais indicam que houve significância. Observa-se na Tabela 1 que os experimentos realizados no conjunto de 50 usuários não apresentaram diferenças significativas para os valores de MSE (com e sem aplicação do contexto) e de MAE (sem aplicação do contexto). No entanto, nos demais experimentos identificaram-se diferenças significativas entre os dados.

Tabela 1. Análise dos dados obtidos nos experimentos realizados

	RMSE Média \pm DP	Valor-p*	MAE Média \pm DP	Valor- p*	MSE Média \pm DP	Valor-p*
50 usuários						
Sem contexto						
Híbrido	0,94 \pm 0,37 ^a	0,002	0,77 \pm 0,32 ^{a,b}	0,112	0,88 \pm 1,09 ^a	0,057
FC	0,87 \pm 0,37 ^a		0,73 \pm 0,31 ^a		0,76 \pm 0,97 ^a	
FBC	1,15 \pm 0,45 ^b		0,93 \pm 0,43 ^b		1,31 \pm 1,50 ^a	
Com contexto						
Híbrido	0,96 \pm 0,37 ^a	< 0,001	0,79 \pm 0,32 ^a	< 0,001	0,93 \pm 1,06 ^a	0,695
FC	0,92 \pm 0,36 ^a		0,75 \pm 0,31 ^a		0,84 \pm 0,96 ^a	
FBC	1,26 \pm 0,48 ^b		1,59 \pm 0,45 ^b		1,05 \pm 1,61 ^a	
100 usuários						
Sem contexto						
Híbrido	0,89 \pm 0,33 ^a	< 0,001	0,74 \pm 0,27 ^a	0,003	0,80 \pm 0,86 ^{a,b}	0,017
FC	0,85 \pm 0,32 ^a		0,71 \pm 0,27 ^a		0,73 \pm 0,78 ^a	
FBC	1,06 \pm 0,46 ^b		0,86 \pm 0,43 ^b		1,12 \pm 1,36 ^b	
Com contexto						
Híbrido	0,94 \pm 0,32 ^a	< 0,001	0,77 \pm 0,28 ^a	< 0,001	0,88 \pm 0,85 ^{a,b}	0,022
FC	0,91 \pm 0,32 ^a		0,74 \pm 0,27 ^a		0,82 \pm 0,78 ^a	
FBC	1,10 \pm 0,45 ^b		0,90 \pm 0,43 ^b		1,20 \pm 1,39 ^b	
150 usuários						
Sem contexto						
Híbrido	0,87 \pm 0,31 ^a	< 0,001	0,72 \pm 0,26 ^a	< 0,001	0,75 \pm 0,77 ^a	0,001
FC	0,84 \pm 0,31 ^a		0,69 \pm 0,25 ^a		0,70 \pm 0,70 ^a	
FBC	1,04 \pm 0,44 ^b		0,85 \pm 0,40 ^b		1,08 \pm 1,27 ^b	
Com contexto						
Híbrido	0,89 \pm 0,31 ^a	< 0,001	0,75 \pm 0,26 ^a	< 0,001	0,80 \pm 0,77 ^a	0,005
FC	0,87 \pm 0,30 ^a		0,72 \pm 0,26 ^a		0,77 \pm 0,71 ^a	
FBC	1,05 \pm 0,44 ^b		0,86 \pm 0,41 ^b		1,10 \pm 1,27 ^b	

(continua)

Tabela 1. Análise dos dados obtidos nos experimentos realizados

(conclusão)						
	RMSE Média ± DP	Valor-p*	MAE Média ± DP	Valor- p*	MSE Média ± DP	Valor-p*
200 usuários						
Sem contexto						
Híbrido	0,86 ± 0,30 ^a	< 0,001	0,71 ± 0,25 ^a	< 0,001	0,74 ± 0,73 ^a	< 0,001
FC	0,84 ± 0,30 ^a		0,69 ± 0,25 ^a		0,70 ± 0,68 ^a	
FBC	1,03 ± 0,45 ^b		0,84 ± 0,40 ^b		1,05 ± 1,21 ^b	
Com contexto						
Híbrido	0,87 ± 0,30 ^a	< 0,001	0,72 ± 0,25 ^a	< 0,001	0,76 ± 0,71 ^a	< 0,001
FC	0,85 ± 0,29 ^a		0,69 ± 0,25 ^a		0,72 ± 0,66 ^a	
FBC	1,03 ± 0,45 ^b		0,85 ± 0,41 ^b		1,06 ± 1,23 ^b	
250 usuários						
Sem contexto						
Híbrido	0,85 ± 0,29 ^a	< 0,001	0,70 ± 0,24 ^a	< 0,001	0,73 ± 0,68 ^a	< 0,001
FC	0,83 ± 0,28 ^a		0,68 ± 0,24 ^a		0,69 ± 0,63 ^a	
FBC	1,02 ± 0,44 ^b		0,84 ± 0,40 ^b		1,05 ± 1,23 ^b	
Com contexto						
Híbrido	0,85 ± 0,28 ^a	< 0,001	0,70 ± 0,24 ^a	< 0,001	0,73 ± 0,66 ^a	< 0,001
FC	0,83 ± 0,28 ^a		0,68 ± 0,24 ^a		0,69 ± 0,62 ^a	
FBC	1,03 ± 0,44 ^b		0,85 ± 0,41 ^b		1,07 ± 1,26 ^b	
300 usuários						
Sem contexto						
Híbrido	0,84 ± 0,29 ^a	< 0,001	0,69 ± 0,24 ^a	< 0,001	0,71 ± 0,68 ^a	< 0,001
FC	0,82 ± 0,28 ^a		0,66 ± 0,24 ^a		0,68 ± 0,64 ^a	
FBC	1,03 ± 0,44 ^b		0,85 ± 0,40 ^b		1,06 ± 1,21 ^b	
Com contexto						
Híbrido	0,86 ± 0,29 ^a	< 0,001	0,70 ± 0,25 ^a	< 0,001	0,74 ± 0,68 ^a	< 0,001
FC	0,83 ± 0,29 ^a		0,68 ± 0,25 ^a		0,69 ± 0,64 ^a	
FBC	1,03 ± 0,44 ^b		0,85 ± 0,40 ^b		1,07 ± 1,22 ^b	
350 usuários						
Sem contexto						
Híbrido	0,85 ± 0,29 ^a	< 0,001	0,69 ± 0,25 ^a	< 0,001	0,72 ± 0,71 ^a	< 0,001
FC	0,83 ± 0,29 ^a		0,66 ± 0,25 ^a		0,68 ± 0,68 ^a	
FBC	1,03 ± 0,45 ^b		0,85 ± 0,41 ^b		1,06 ± 1,26 ^b	
Com contexto						
Híbrido	0,85 ± 0,30 ^a	< 0,001	0,70 ± 0,25 ^a	< 0,001	0,73 ± 0,71 ^a	< 0,001
FC	0,83 ± 0,30 ^a		0,67 ± 0,25 ^a		0,69 ± 0,69 ^a	
FBC	1,04 ± 0,45 ^b		0,85 ± 0,42 ^b		1,08 ± 1,28 ^b	

FC representa Filtragem Colaborativa e FBC representa Filtragem Baseada em Conteúdo

*Valor obtido após aplicação do teste de análise de variância ANOVA de uma via.

^{a,b}Letras distintas representam diferenças estatisticamente significativas após aplicação do *post hoc* teste de Tukey ($p < 0,05$).

Fonte: Dados da pesquisa (2020).

Na Tabela 2 observa-se que o sistema de recomendação híbrido apresentou os maiores valores para as métricas quando comparados aos da filtragem colaborativa. Assim, considerando a literatura que indica quanto menor o valor (mais próximo de zero) melhor é o modelo, pode-se concluir que nos experimentos realizados a filtragem colaborativa quando aplicada individualmente foi melhor que a híbrida. Baseando-se nos valores das métricas apresentadas para os dois tipos de filtragem pode-se verificar as maiores diferenças para RMSE de 0,09, MSE de 0,16 e MAE de 0,06, indicando baixa variação nos valores.

Tabela 2. Híbrido e Filtragem Colaborativa

	Híbrido				Filtragem Colaborativa		
	RMSE	MSE	MAE		RMSE	MSE	MAE
50	0,962	0,925	0,794	50	0,869	0,756	0,730
100	0,938	0,881	0,768	100	0,853	0,727	0,711
150	0,894	0,799	0,746	150	0,837	0,700	0,691
200	0,871	0,759	0,716	200	0,837	0,700	0,688
250	0,853	0,727	0,701	250	0,829	0,687	0,678
300	0,858	0,736	0,700	300	0,824	0,679	0,665
350	0,852	0,726	0,696	350	0,827	0,683	0,664

Fonte: Dados da pesquisa (2020).

A Tabela 3 apresenta os menores valores das métricas probabilísticas empregadas para o sistema de recomendação híbrido, quando comparados ao da filtragem baseada em conteúdo. Assim, pode-se concluir que nos experimentos realizados a filtragem híbrida foi melhor que a baseada em conteúdo. Baseando-se nos valores das métricas apresentadas para os dois tipos de filtragem pode-se verificar as maiores diferenças para RMSE de 0,18, MSE de 0,38 e MAE de 0,15, as quais foram maiores que as obtidas entre a híbrida e a colaborativa.

Tabela 3. Híbrido e Filtragem Baseada em Conteúdo

	Híbrido				Filtragem Baseada em Conteúdo		
	RMSE	MSE	MAE		RMSE	MSE	MAE
50	0,962	0,925	0,794	50	1,145	1,314	0,933
100	0,938	0,881	0,768	100	1,059	1,121	0,862
150	0,894	0,799	0,746	150	1,037	1,075	0,848
200	0,871	0,759	0,716	200	1,026	1,054	0,835
250	0,853	0,727	0,701	250	1,024	1,049	0,843
300	0,858	0,736	0,700	300	1,030	1,060	0,846
350	0,852	0,726	0,696	350	1,030	1,060	0,852

Fonte: Dados da pesquisa (2020).

Mediante as análises apresentadas, observa-se que os resultados do sistema de recomendação híbrido proposto foram menos precisos que os da filtragem colaborativa, porém com pouca significância, visto que a maior diferença RMSE foi de 0,09. No entanto, a filtragem híbrida obteve melhores resultados que a filtragem baseada em conteúdo, com uma diferença máxima de 0,18 para o RMSE. Assim, ao considerar que as avaliações foram de zero a cinco, essas diferenças são baixas.

Considerando-se que foi aplicado para a hibridização o método de acréscimo de características, o qual permite a recomendação de forma individual e sequencial, os resultados obtidos (Tabelas 2 e 3) possibilitaram identificar que a técnica baseada em conteúdo influenciou negativamente os resultados do sistema de recomendação híbrido, diminuindo a sua precisão.

A menor precisão nos resultados da filtragem baseada em conteúdo pode ter ocorrido devido a utilização de uma única forma de identificação de similaridade entre os filmes, que no caso foi por meio da comparação entre os gêneros. Os filmes podem apresentar mais de um gênero, porém aqueles que pertencem a poucos gêneros são caracterizados como similares, o que acontece na base MovieLens, causando uma

generalização, já que pode existir uma quantidade significativa de filmes com os mesmos gêneros. Por exemplo, filmes do gênero comédia podem ter características importantes que os diferenciam, influenciando no gosto e na possível nota atribuída pelo usuário. O Gráfico 2, mostra que 86% dos filmes da base MovieLens têm três ou menos gêneros, representando 8407 filmes em uma distribuição de apenas 20 gêneros. Uma integração da base MovieLens com algum serviço de dados de filmes como The Movie Database (TMDB) e Internet Movie Database (IMDB) pode auxiliar na identificação da similaridade entre filmes a partir de outras características, como diretor e atores principais.

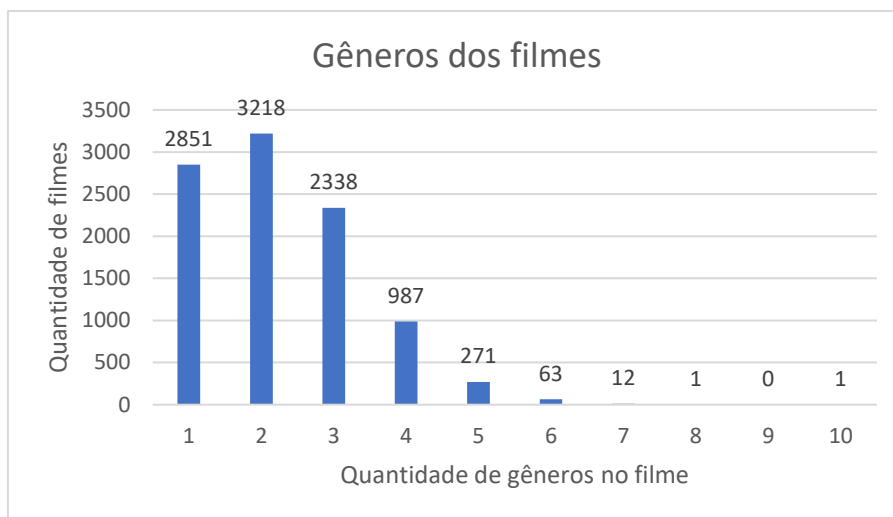


Gráfico 2. Gêneros dos filmes

Os resultados do sistema híbrido proposto podem ser considerados bons para recomendação devido aos baixos valores gerados para as métricas utilizadas. Os maiores valores obtidos foram para o RMSE de 0,962, o MSE de 0,925 e o MAE de 0,794, isso representa que em uma recomendação a variação entre a nota que o usuário atribuiria, calculada pelo sistema, e a real nota avaliada tende a ser inferior a 1. Dessa forma, em um filme que o sistema calculou a atribuição de uma nota 5, o usuário tende a avaliar com uma nota não menor a 4. Sendo assim, o sistema de recomendação proposto tende a não atribuir uma nota máxima a um filme que o usuário avaliou com a nota mínima, bem como o contrário.

A fim de identificar o desempenho da pré-filtragem contextual, nos resultados das recomendações, são apresentados os valores das métricas RMSE, MSE e MAE para a aplicação do contexto nos modelos desenvolvidos nesta pesquisa, sendo eles a filtragem híbrida, a colaborativa e a baseada em conteúdo.

A Tabela 4 apresenta a influência da pré-filtragem contextual nos resultados das técnicas de recomendação desenvolvidas. Apenas no conjunto de 250 usuários da filtragem híbrida a aplicação da pré-filtragem contextual melhorou os resultados das três métricas. Ao comparar os resultados com e sem aplicação de contexto, observa-se que as maiores diferenças foram para o RMSE de 0,12 na filtragem baseada em conteúdo, MSE de 0,1 na filtragem colaborativa e MAE de 0,66 na filtragem baseada em conteúdo. Considerando-se, que a nota máxima que um usuário pode atribuir a um filme é 5, e este representa o valor máximo a ser obtido nas três métricas (em caso de não concordância entre o usuário e o sistema), a diferença entre os resultados com e sem contexto é baixa.

No entanto, apesar da diferença baixa entre os valores obtidos, observou-se nesta pesquisa que o contexto influenciou negativamente nos resultados, o que pode ter ocorrido devido a utilização de apenas um contexto, se era fim de semana ou não.

A base MovieLens, mesmo sendo uma das mais completas e com grande quantidade de dados sobre filmes, apresentou limitação para aplicação da pré-filtragem contextual nesta pesquisa, visto que a informação disponível sobre o contexto dos usuários é a data e o horário da avaliação.

Tabela 4. Aplicação da pré-filtragem contextual

Usuários	COM APLICAÇÃO DE CONTEXTO			SEM APLICAÇÃO DE CONTEXTO		
	RMSE	MSE	MAE	RMSE	MSE	MAE
Filtragem híbrida						
50	0,962	0,925	0,794	0,940	0,883	0,765
100	0,938	0,881	0,768	0,893	0,797	0,737
150	0,894	0,799	0,746	0,867	0,751	0,719
200	0,871	0,759	0,716	0,862	0,743	0,712
250	0,853	0,727	0,701	0,855	0,731	0,703
300	0,858	0,736	0,700	0,841	0,708	0,691
350	0,852	0,726	0,696	0,848	0,719	0,689
Filtragem colaborativa						
50	0,918	0,842	0,747	0,869	0,756	0,730
100	0,907	0,823	0,736	0,853	0,727	0,711
150	0,875	0,765	0,719	0,837	0,700	0,691
200	0,850	0,723	0,694	0,837	0,700	0,688
250	0,832	0,692	0,679	0,829	0,687	0,678
300	0,832	0,692	0,854	0,824	0,679	0,665
350	0,829	0,687	0,674	0,827	0,683	0,664
Filtragem baseada em conteúdo						
50	1,263	1,053	1,595	1,145	1,314	0,933
100	1,097	1,204	0,905	1,059	1,121	0,862
150	1,047	1,096	0,863	1,037	1,075	0,848
200	1,031	1,064	0,854	1,026	1,054	0,835
250	1,035	1,070	0,853	1,024	1,049	0,843
300	1,035	1,070	0,854	1,030	1,060	0,846
350	1,038	1,078	0,855	1,030	1,060	0,852

Fonte: Dados da pesquisa (2020).

O conjunto de dados MovieLens apresentou bom desempenho na filtragem colaborativa sem aplicação da pré-filtragem contextual. Porém, na filtragem baseada em conteúdo seu desempenho ficou limitado as poucas informações disponíveis dos filmes (gêneros e ano de lançamento), o que poderia ser diferente se na base tivessem mais informações. A utilização do MovieLens na pré-filtragem contextual se mostrou limitada, pois se tinha disponível somente a data e horário da avaliação, limitando o entendimento do contexto em que o usuário se encontrava. Mediante o exposto, a aplicação de um conjunto de dados com diferentes identificadores do contexto dos usuários parece ser o mais recomendado para experimentos utilizando a pré-filtragem contextual.

5. Discussão

Os valores obtidos indicaram bons resultados para utilização do sistema de recomendação híbrido proposto, apresentando índices coerentes para recomendação de filmes. Os valores RMSE obtidos se assemelham aos do estudo realizado por Do et al (2020), variando próximo a 1. Esperava-se resultados mais precisos em relação aos que foram obtidos nesta pesquisa, o sistema de recomendação híbrido se mostrou levemente inferior

a técnica colaborativa, influenciado diretamente pela técnica baseada em conteúdo que não obteve o desempenho esperado devido a forma como foi empregado.

A técnica de hibridismo, integrando as filtragens colaborativa e baseada em conteúdo, pôde ser aplicada com sucesso. Os resultados do sistema híbrido foram menos precisos que os da filtragem colaborativa, porém melhores que os da baseada em conteúdo. A inferioridade nos resultados do hibridismo em relação a colaborativa foi causada pela técnica baseada em conteúdo, pois o método de acréscimo de características empregado no sistema híbrido utiliza as soluções de ambas as filtragens sem diferenciais, o que causa uma influência direta na precisão das recomendações. Isso ocorre também na pesquisa de Bezerra et al. (2018), tendo-se a filtragem baseada em conteúdo influenciando diretamente nos resultados, visto que muitos usuários utilizaram mais conteúdo do Youtube as recomendações se limitaram a vídeos, não recomendando outros meios. Na pesquisa de Do et al (2020) esse problema não ocorre, pois emprega o método de hibridismo de ponderação de cada resultado, atribuindo-se a técnica que se tem mais confiança um valor maior na recomendação.

Nesta pesquisa, a aplicação do contexto, obtido por meio dos dados contidos na base MovieLens, não proporcionou melhorias nos resultados, contrapondo os achados de Kristoffersen et al. (2020) e corroborando o estudo de Avancini (2016) em que foram utilizados poucos parâmetros contextuais. Kristoffersen et al. (2020) utilizou-se de uma pesquisa específica com centenas de pessoas e um período de cinco semanas, enquanto Avancini (2016) obteve o contexto por meio de uma aplicação desenvolvida. Assim, entende-se que a maior disponibilidade de informações contextuais dos usuários, favorecem a análise e utilização do contexto nos sistemas de recomendação.

6. Conclusão

Nesta pesquisa foi desenvolvido um sistema de recomendação híbrido utilizando a filtragem colaborativa e a baseada em conteúdo com a aplicação da pré-filtragem contextual. O hibridismo foi avaliado por meio das métricas probabilísticas RMSE, MSE e MAE, comparando individualmente as recomendações fornecidas com as das técnicas de filtragem colaborativa e baseada em conteúdo. O sistema de recomendação desenvolvido foi integrado a um aplicativo móvel destinado a sugerir os filmes por meio da filtragem híbrida proposta.

A pesquisa apresentou a aplicação destas técnicas em sistemas de recomendação híbridos e a sua avaliação por meio de medidas probabilísticas. O hibridismo de ambas as técnicas de filtragem baseada em conteúdo e colaborativa pelo método de acréscimo de características foi desenvolvido com sucesso para a recomendação de filmes. O algoritmo híbrido se mostrou viável para as recomendações, pois as métricas probabilísticas apresentaram uma variação menor que 1 entre a avaliação real do filme realizada pelo usuário e a calculada pelo sistema de recomendação.

O algoritmo KNN foi empregado com sucesso na filtragem colaborativa, assim como no sistema híbrido desenvolvido, possibilitando a identificação da similaridade entre os usuários da aplicação. O conjunto de dados referente a filmes, MovieLens, apresentou nesta pesquisa limitações na aplicação da técnica de pré-filtragem contextual causada pela baixa quantidade de dados sobre o contexto dos usuários. Também, na aplicação da filtragem baseada em conteúdo os dados para identificação da similaridade entre os filmes foram limitados, visto que só foi possível a utilização dos gêneros para

identificar se um filme é semelhante a outro, enquanto para a realização da colaborativa os dados deste conjunto se mostraram adequados.

O sistema de recomendação híbrido desenvolvido obteve melhores resultados nos valores RMSE, MSE e MAE que a baseada em conteúdo, porém a colaborativa se mostrou mais precisa, o que pode ser atribuído a menor precisão da técnica baseada em conteúdo, que influenciou os dados negativamente na técnica híbrida. O RMSE obtido pelo algoritmo híbrido desta pesquisa variou entre 0,852 e 0,962, sendo similar a valores de outras pesquisas de sistemas de recomendação que variam entre 0,7 e 1,3.

Em pesquisas futuras recomenda-se a utilização de uma base de dados com informações detalhadas sobre os filmes para aplicação da similaridade, sendo também possível identificar o contexto do usuário no momento da avaliação de um filme, como por exemplo, se está acompanhado e se estava assistindo a algum programa anteriormente. Também, a integração com redes sociais e análises de sentimentos podem auxiliar na identificação do contexto do usuário. Outras possibilidades de pesquisas na área consistem no emprego do *deep learning* em sistemas de recomendação e de métodos variados de hibridismo, como por exemplo, o método ponderado.

Referências

- Adomavicius, G., Ricci, F., Tuzhilin, A. e Tuzhilin, A. (2011). Context-Aware Recommender Systems. *Association for the Advancement of Artificial Intelligence*, p. 67–80.
- Al-Ghuribi, S. M. e Mohd Noah, S. A. (2019). Multi-Criteria Review-Based Recommender System-The State of the Art. *IEEE Access*, v. 7, p. 169446–169468.
- Avancini, V. P. (2016). Um Sistema De Recomendação Sensível Ao Contexto Para Atividades De Lazer. p. 1–77.
- Barbosa, C. E. M. (2014). Estudo de Técnicas de Filtragem Híbrida em Sistemas de Recomendação de Produtos. p. 1–84.
- Bezerra, S. F., Costa, A., Neto, F. M., Silva, P. e Monteiro, B. D. S. (2018). Um Sistema de Recomendação Híbrido Integrado a Ontologia em um Ambiente de Aprendizagem Ubíqua. *Anais do XXIX Simpósio Brasileiro de Informática na Educação (SBIE 2018)*, v. 1, n. Cbie, p. 1253–1262.
- Bobadilla, J., Hernando, A., Ortega, F. e Bernal, J. (2011). A framework for collaborative filtering recommender systems. *Expert Systems with Applications*, v. 38, n. 12, p. 14609–14623.
- Burke, R. (2002). Hybrid Recommender Systems : Survey and Experiments †. p. 1–29.
- Chen, R., Hua, Q., Chang, Y. S., et al. (2018). A survey of collaborative filtering-based recommender systems: from traditional methods to hybrid methods based on social networks. *IEEE Access*, v. 6, p. 64301–64320.
- Do, H. Q., Le, T. H. e Yoon, B. (feb 2020). Dynamic Weighted Hybrid Recommender Systems. *International Conference on Advanced Communication Technology, ICACT*, v. 2020, p. 644–650.
- Fathoni, K., Zikky, M., Nurhayati, A. S. e Prasetyaningrum, I. (2018). Application of K-Nearest Neighbor Algorithm For Puzzle Game of Human Body's System Learning on

- Virtual Mannequin. *2018 International Conference on Applied Science and Technology (iCAST)*, p. 530–535.
- Fessahaye, F., Perez, L., Zhan, T., et al. (2019). T-RECSYS: A Novel Music Recommendation System Using Deep Learning. *2019 IEEE International Conference on Consumer Electronics, ICCE 2019*, p. 18–23.
- Gatzioura, A., Vinagre, J., Jorge, A. M. e Sánchez-Marrè, M. (jul 2019). A Hybrid Recommender System for Improving Automatic Playlist Continuation. *IEEE Transactions on Knowledge and Data Engineering*, v. 4347, n. JULY, p. 1–12.
- Huynh, H. X., Phan, N. Q., Pham, N. M., et al. (2020). Context-Similarity Collaborative Filtering Recommendation. *IEEE Access*, v. 8, p. 33342–33351.
- Irfan, R., Khalid, O., Usman, M. e Khan, S. (2019). SocialRec: A Context-Aware Recommendation Framework With Explicit Sentiment Analysis. *IEEE Access*, v. 7, p. 116295–116308.
- Kim, B. M., Li, Q., Park, C. S., Kim, S. G. e Kim, J. Y. (2006). A new approach for combining content-based and collaborative filters. *Journal of Intelligent Information Systems*, v. 27, n. 1, p. 79–91.
- Kristoffersen, M. S., Shepstone, S. E. e Tan, Z. H. (2020). The importance of context when recommending TV Content: Dataset and algorithms. *IEEE Transactions on Multimedia*, v. 22, n. 6, p. 1531–1541.
- LÁZARO, A. D. S. (2010). Análise e seleção de algoritmos de filtragem de informação para solução do problema cold-start item.
- Park, D. H., Kim, H. K. e Kim, J. K. (2010). A Literature Review and Classification of Recommender Systems on Academic Journals. p. 139–152.
- Peireira, A. B. (2016). Sistemas De Recomendação Baseados Em Contexto Físico E Social. *Centro de informática UFPE*, p. 1–66.
- Primo, T. T. (2013). Método de representação de conhecimento baseado em ontologias para apoiar sistemas de recomendação educacionais. p. 120.
- Santana, L. L. B. de S. (2018). Explorando Relações entre Usuários em um Sistema de Recomendação Híbrido Baseado em Filmes.
- Schafer, J. Ben, Frankowski, D., Herlocker, J. e Sen, S. (2007). Collaborative filtering recommender systems. In: (Ed.). *The adaptive web. Springer-Verlag Berlin Heidelberg*, n. January 2007, p. 291–324.
- Seyednezhad, S. M. M., Cozart, K. N., Brollan, J. A. e Smith, A. O. (2018). A Review on Recommendation Systems: Context-aware to Social-based. n. November.
- Shaikh, S., Rathi, S. e Janrao, P. (2017). Graph Based Approached. *2017 IEEE 7th International Advance Computing Conference*, p. 932–935.
- Shardanand, U. e Maes, P. (1995). Social information filtering. p. 210–217.
- Sharma, A. e Kaur, P. (2019). A Multitenant Data Store Using a Column Based NoSQL Database. *2019 12th International Conference on Contemporary Computing, IC3 2019*, n. iii.

- Wazlawick, R. S. (2014). *Metodologia de Pesquisa para Ciência da Computação*. Elsevier Editora Ltda, 2 ed.
- Yadav, R. (2018). A Recommendation System for E-Commerce Base on Client Profile. *2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)*, n. Icoei, p. 1316–1322.
- Yamaguti, Renan de Oliveira, J. M. A. C. (sep 2013). Algoritmo Híbrido Para Sistemas De Recomendação Utilizando Filtragem Colaborativa E Algoritmo. *Anais do III Encontro de Iniciação em Desenvolvimento Tecnológico e Inovação*,
- Yang, J., Yang, C. e Hu, X. (2016). A Study of Hybrid Recommendation Algorithm Based on User. *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, v. 02, n. 2, p. 261–264.
- Yin, H., Cui, B., Chen, L., Hu, Z. e Zhou, X. (2014). Dynamic User Modeling in Social Media Systems. *ACM Transactions on Information Systems*, v. 5, n. 212.