

UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO"
FACULDADE DE CIÊNCIAS - CAMPUS BAURU
DEPARTAMENTO DE COMPUTAÇÃO
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

GUILHERME SOUZA MINGRONI

**SISTEMA DE RECOMENDAÇÃO DE FILMES BASEADO EM
FILTRAGEM**

BAURU
Novembro/2024

M664s

Mingroni, Guilherme Souza

SISTEMA DE RECOMENDAÇÃO DE FILMES BASEADO EM
FILTRAGEM / Guilherme Souza Mingroni. -- Bauru, 2024
48 p.

Trabalho de conclusão de curso (Bacharelado - Ciência da
Computação) - Universidade Estadual Paulista (UNESP), Faculdade
de Ciências, Bauru

Orientador: Leandro Passos

1. Catálogo de filmes. 2. Ciência da computação.. 3. Sistemas de
computação interativos. 4. Python (Computer program language). I.
Título.

Guilherme Souza Mingroni

SISTEMA DE RECOMENDAÇÃO DE FILMES BASEADO EM FILTRAGEM

Trabalho de Conclusão de Curso do Curso
de Bacharelado em Ciência da Computa-
ção da Universidade Estadual Paulista "Jú-
lio de Mesquita Filho", Faculdade de Ciên-
cias, Campus Bauru.

Banca Examinadora

Prof. Leandro Passos Aparecido Junior

Orientador

Universidade Estadual Paulista "Júlio de
Mesquita Filho"
Faculdade de Ciências
Departamento de Computação

**Prof^a. Dr^a. Simone das Graças
Domingues Prado**

Universidade Estadual Paulista "Júlio de
Mesquita Filho"
Faculdade de Ciências
Departamento de Computação

**Prof. Dr. Kelton Augusto Pontara da
Costa**

Universidade Estadual Paulista "Júlio de
Mesquita Filho"
Faculdade de Ciências

Bauru, 12 de Novembro de 2024.

Dedico este trabalho a minha família, que sempre foram meu suporte e fonte de inspiração, e a todas as pessoas (amigos e muito a minha namorada), que tornaram esta jornada mais leve e mais tranquila para ser percorrida.

Agradecimentos

Agradeço, primeiramente, a Deus, por me dar forças e sabedoria durante toda essa jornada. Ele me mostrou que era possível ultrapassar barreiras que jamais imaginei que conseguiria. Aos meus pais, pelo apoio incondicional e pelas orientações que sempre me guiaram, sem a ajuda deles não teria a oportunidade de ter vivenciado essa jornada. Agradeço a minha irmã Julia, por apoiar meus pais durante o período que estive fora e protegê-los, além de torcer por mim todos esses anos. Agradeço ao meu orientador, Leandro Passos, pela paciência, pelos conselhos valiosos e pela orientação ao longo do desenvolvimento deste trabalho. Agradeço também aos demais professores do curso, principalmente a prof Andréa Carla Gonçalves Vianna, que contribuíram para a minha formação com seus ensinamentos e experiências, além de todo o apoio e incentivo para continuar aprendendo, mesmo em tempos difíceis como a pandemia. Aos meus amigos e colegas, pela companhia, pela troca de ideias, pelos conselhos e pelo apoio em momentos desafiadores. Por fim, não menos importante, agradeço a minha namorada, Amanda Souza Reche, pelo amor e compreensão, por lutar ao meu lado algumas minhas batalhas e sempre estar me apoiando, principalmente nessa fase final de curso. A todos vocês, meu sincero obrigado.

Resumo

Este trabalho aborda a evolução e a relevância dos sistemas de recomendação no contexto atual de consumo de mídia, especialmente em plataformas de streaming. Com o advento da inteligência artificial e do aprendizado de máquina, os sistemas de recomendação se tornaram essenciais para personalizar a experiência do usuário, mas ainda enfrentam desafios como o filtro bolha e a diversidade nas sugestões. O sistema proposto, chamado FilmMatch, busca superar essas limitações por meio da utilização de múltiplas métricas de similaridade, incluindo Similaridade Cosseno, Correlação de Pearson e Índice de Jaccard. A pesquisa se propõe a desenvolver e avaliar um sistema de recomendação de filmes eficaz, analisando a eficácia das métricas e comparando seu desempenho com outros sistemas existentes. Os resultados indicam que, embora a precisão do sistema esteja dentro da média, o recall é uma área crítica que precisa de melhorias. Este trabalho contribui para a discussão sobre a importância de recomendações diversificadas e personalizadas no setor de entretenimento.

Palavras-chave: Sistemas de Recomendação; Consumo de Mídia, Plataformas de Streaming; Métricas de similaridade

Abstract

This work addresses the evolution and relevance of recommendation systems in the current media consumption context, especially on streaming platforms. With the advent of artificial intelligence and machine learning, recommendation systems have become essential for personalizing user experience, yet they still face challenges such as the filter bubble and suggestion diversity. The proposed system, called FilmMatch, aims to overcome these limitations by utilizing multiple similarity metrics, including Cosine Similarity, Pearson Correlation, and Jaccard Index. The research aims to develop and evaluate an effective movie recommendation system, analyzing the effectiveness of metrics and comparing its performance with existing systems. Results indicate that while the system's precision is within average range, recall is a critical area needing improvement. This work contributes to the discussion on the importance of diversified and personalized recommendations in the entertainment sector.

Keywords: Recommendation Systems; Media Consumption; Streaming Platforms; Similarity Metrics.

Lista de figuras

Figura 1 – Filtragem Baseada em conteúdo	20
Figura 2 – Clusterização	25
Figura 3 – Funcionamento do DBSCAN	25
Figura 4 – Plataforma FilmMatch	33
Figura 5 – Painel de opções das métricas do FilmMatch	33
Figura 6 – Etapas de Processamento dos dados	35
Figura 7 – Código de importação das bases	35
Figura 8 – Código de Seleção das variáveis e renomeação de parte das bases	36
Figura 9 – Código de União das bases	36
Figura 10 – Top 5 - correlação de pearson	40
Figura 11 – Top 5 - similaridade total	41

Lista de tabelas

Tabela 1 – Comparação de métodos de recomendação	18
Tabela 2 – Comparação de Precisão entre Sistemas	43
Tabela 3 – Comparação de Recall entre Sistemas	43

Lista de abreviaturas e siglas

MPA *Motion Pictures Association*

Sumário

1	INTRODUÇÃO	13
1.1	Diferenciais do Sistema Proposto	14
1.2	Objetivos	14
1.2.1	Objetivo Geral	14
1.2.2	Objetivos Específicos	15
1.3	Justificativa	15
1.4	Pergunta da Pesquisa	15
1.5	Organização do Trabalho	16
2	TRABALHOS RELACIONADOS	17
3	FUNDAMENTAÇÃO TEÓRICA	19
3.1	Evolução dos Sistemas de Recomendação de Filmes	19
3.1.1	Filtragem Baseada em Conteúdo	20
3.1.2	Filtragem Colaborativa	20
3.1.2.1	Filtragem Colaborativa Baseada em usuários	21
3.1.2.2	Filtragem Colaborativa Baseada em Itens	21
3.1.3	Sistemas de Recomendação Híbridos	21
3.2	Métricas de Similaridade	22
3.2.1	Similaridade Cosseno	22
3.2.2	Correlação de Pearson	23
3.2.3	Índice de Jaccard	23
3.2.4	Métrica Combinada	24
3.3	Clusterização	24
3.3.1	DBSCAN	25
3.4	Métricas de avaliação	26
3.4.1	Precisão: Definição e Cálculo	26
3.4.2	Recall: Definição e Cálculo	26
4	MODELO PROPOSTO	28
4.1	Filtragem Baseada em Conteúdo	28
4.1.1	Índice de Jaccard	28
4.1.2	Clusterização por Gênero e Diretores	28
4.2	Filtragem Colaborativa	29
4.2.1	Correlação de Pearson	29
4.2.2	Similaridade Cosseno	29

4.2.3	Filtragem Híbrida (Similaridade Total)	29
4.2.4	Funcionamento do Modelo	30
5	METODOLOGIA	31
5.1	Ambiente e Ferramentas	31
5.1.1	Google Colab	31
5.1.2	Python	31
5.1.2.1	Bibliotecas utilizadas	31
5.1.2.1.1	Pandas	31
5.1.2.1.2	Numpy	32
5.1.2.1.3	Scikit-Learn	32
5.1.2.1.4	Ipywidgets	32
5.2	Datasets	32
5.2.1	Netflix Movies and TV Shows	33
5.2.2	IMDB Movies Dataset	34
5.2.3	TMDB Movies (900k movies + daily uptades)	34
5.3	Etapas de Processamento	34
5.3.1	Importação das bases	35
5.3.2	Seleção e Renomeação das Variáveis	36
5.3.3	União dos Dados	36
5.3.4	Limpeza e Transformação	37
5.3.4.1	Remoção de Valores Nulos e Zerados	37
5.3.4.2	Transformação das Variáveis Categóricas em Colunas Binárias (One-Hot Encoding)	37
5.3.4.3	Conversão de Variáveis de Contagem de Votos de Float para Int das Variáveis Categóricas em Colunas Binárias (One-Hot Encoding)	37
5.3.5	Implementação das Funções de Similaridade	38
5.3.5.1	Índice de Jaccard	38
5.3.5.2	Similaridade Cosseno	38
5.3.5.3	Correlação de Pearson	39
5.3.5.4	DBSCAN	39
5.3.5.5	Similaridade Total	39
6	RESULTADOS	40
6.1	Recomendações de filmes	40
6.2	Aproximação e Tempo Real do Tempo de Execução por Métrica	41
6.2.1	Índice de Jaccard	41
6.2.2	Similaridade Cosseno	41
6.2.3	Correlação de Pearson	41
6.2.4	Clusterização (DBSCAN)	42

6.2.5	Similaridade Total	42
6.3	Resultados de Precisão	42
6.3.1	Resultados de Precisão	42
6.3.2	Resultados de Recall	43
7	CONCLUSÃO	44
7.1	Perspectivas futuras	44
7.1.1	Implementação de novos algoritmos e interligação de métricas . . .	44
7.1.2	Aplicação de outros métodos de avaliação	45
7.1.3	Expansão para plataformas integradas	45
7.1.4	Integração com APIs de atualização de conteúdo	45
7.1.5	Detalhamento de atualizações	45
	Referências	46

1 Introdução

Desde os primórdios do cinema, o entretenimento audiovisual tem sido uma parte essencial da vida moderna, moldando nossa cultura e oferecendo uma janela para diferentes realidades e experiências (THOMPSON E BORDWELL 1994). Ao longo dos anos, o cinema evoluiu, desde os primeiros filmes mudos até as grandes produções de Hollywood, elaboradas por empresas como Warner, Marvel, Disney, entre outras. Paralelamente, o campo da Inteligência Artificial (IA) começou a despontar no século XX, com os primeiros passos dados por pesquisadores como Alan Turing, que em 1950 propôs o famoso "Teste de Turing", um marco para avaliar a inteligência de máquinas (Turing 1950). À medida que a computação e as técnicas de aprendizado de máquina se desenvolveram, uma intersecção entre o entretenimento e a tecnologia se consolidou, principalmente no século XXI (Russell e Norvig 2016).

A explosão da internet e o advento dos serviços de streaming revolucionaram a forma como consumimos filmes e programas de TV. Netflix, fundada em 1998 como uma plataforma de aluguel de DVDs, transformou-se em um gigante do streaming, marcando o início da era do entretenimento on-demand. Ao mesmo tempo, os avanços no campo do aprendizado de máquina e da IA abriram novas possibilidades para a personalização de conteúdo e a criação de sistemas de recomendação cada vez mais sofisticados. A ascensão de plataformas como Amazon Prime Video em 2006 e o uso de algoritmos complexos de aprendizado profundo marcaram uma nova fase para os sistemas de recomendação. Koren et al (2009) destacaram essa transição para métodos mais complexos, como redes neurais e análise de dados em larga escala.

Atualmente, os sistemas de recomendação são peças centrais em plataformas de *streaming* como Netflix, Amazon Prime Video, Disney+ e HBO Max, oferecendo aos usuários recomendações personalizadas em meio a um vasto catálogo de opções. No entanto, esses sistemas ainda enfrentam desafios importantes, como o problema do filtro bolha, onde o usuário acaba recebendo sugestões limitadas e repetitivas (Nguyen et al. 2014), e a dificuldade em lidar com a diversidade de gostos dos usuários. Embora esses sistemas tenham se tornado indispensáveis para orientar os usuários em um mar de opções, ainda há espaço para melhorias em termos de precisão e diversidade das recomendações.

Ao analisar o campo dos sistemas de recomendação, percebe-se como este evoluiu consideravelmente desde as primeiras abordagens baseadas em filtragem colaborativa. Originalmente, essas técnicas utilizavam o comportamento de usuários semelhantes para sugerir filmes ou produtos. Com o tempo, foram integradas aborda-

gens baseadas em conteúdo, onde os itens recomendados eram sugeridos com base nas características de filmes ou programas previamente consumidos pelo usuário. No entanto, ambos os métodos enfrentam limitações, como a tendência a repetir recomendações semelhantes, reforçando o comportamento passado do usuário, o que contribui para a criação do filtro bolha. Além disso, a incapacidade de capturar nuances das preferências pessoais e a dificuldade em equilibrar diversidade com precisão continuam a ser problemas centrais.

Considerando a relevância dos sistemas de recomendação no contexto atual, este trabalho visa apresentar uma alternativa que ultrapasse algumas das limitações dos métodos tradicionais, como o filtro bolha e a baixa diversidade nas sugestões. Para isso, é proposto um sistema de recomendação, chamado FilmMatch fundamentado em múltiplas métricas de similaridade, com o objetivo de oferecer uma experiência de recomendação mais flexível e ajustável às preferências individuais dos usuários.

1.1 Diferenciais do Sistema Proposto

O sistema de recomendação proposto neste trabalho busca superar algumas dessas limitações por meio da introdução de múltiplas métricas de similaridade, dando ao usuário a flexibilidade de escolher entre diferentes abordagens matemáticas para calcular a proximidade entre os filmes e suas preferências pessoais. As métricas utilizadas incluem Similaridade Cosseno, Correlação de Pearson, Índice de Jaccard e técnicas de Clusterização. Esse diferencial permite que o sistema ofereça uma gama mais diversificada de recomendações, adaptando-se às necessidades e gostos individuais do usuário.

A implementação dessas métricas no sistema permite maior personalização e variedade nas recomendações, reduzindo a probabilidade de o usuário ficar preso em um ciclo repetitivo de sugestões. Além disso, a técnica de clusterização possibilita a criação de grupos de filmes com características semelhantes, otimizando a diversidade de recomendações e evitando o efeito de filtro bolha. O objetivo é oferecer um sistema de recomendação que combine precisão com diversidade, garantindo uma experiência de uso mais rica e satisfatória.

1.2 Objetivos

1.2.1 Objetivo Geral

Desenvolver e avaliar um sistema de recomendação de filmes eficaz, capaz de oferecer sugestões personalizadas e relevantes para os usuários, contribuindo para melhorar a experiência do usuário em plataformas de streaming de vídeo.

1.2.2 Objetivos Específicos

- Realizar uma revisão da literatura sobre técnicas e algoritmos de sistemas de recomendação de filmes, destacando as abordagens mais relevantes e eficazes.
- Coletar e pré-processar dados de avaliações de filmes e informações de usuários para construir um conjunto de dados representativo.
- Desenvolver e implementar um sistema de recomendação de filmes utilizando técnicas de filtragem colaborativa e baseada em conteúdo, bem como técnicas de aprendizado de máquina.
- Avaliar o sistema utilizando métricas de similaridade como Similaridade de Cosseno, Correlação de Pearson, Índice de Jaccard, além de técnicas de clusterização.
- Comparar o desempenho do sistema com outros sistemas de recomendação existentes no cenário.
- Analisar os resultados obtidos e fornecer recomendações para melhorias futuras.

1.3 Justificativa

O desenvolvimento de sistemas de recomendação mais eficazes e personalizados é crucial no contexto atual de consumo de mídia, principalmente diante do aumento exponencial do uso de plataformas de streaming. Segundo dados da *Motion Pictures Association* (MPA), em 2020, o número de assinantes de serviços de streaming de vídeo ultrapassou 1 bilhão, representando um crescimento de 26 por cento em relação ao ano anterior. No Brasil, o consumo de streaming também é massivo, sendo o segundo país que mais utiliza esses serviços.

Além disso, a personalização tem um impacto direto na satisfação dos usuários e no sucesso das plataformas. Estudos mostram que recomendações baseadas em algoritmos influenciam diretamente a escolha dos usuários, com até 50 por cento dos vídeos assistidos em plataformas como TikTok sendo recomendados por esses sistemas. Isso demonstra a importância dos sistemas de recomendação não apenas para melhorar a experiência do usuário, mas também para aumentar a retenção e fidelidade às plataformas.

1.4 Pergunta da Pesquisa

A pergunta central que orienta este trabalho é: É possível oferecer sugestões mais diversificadas e otimizadas aos usuários através da utilização de diferentes

métricas de similaridade no processo de filtragem colaborativa? A ideia é avaliar se a utilização de diferentes métricas de similaridade e técnicas de clusterização permitirá gerar recomendações mais precisas e diversificadas, oferecendo uma experiência superior em relação aos sistemas tradicionais.

1.5 Organização do Trabalho

O restante deste trabalho apresenta uma análise dos sistemas de recomendação de filmes, propondo o sistema FilmMatch como uma alternativa inovadora às limitações dos métodos tradicionais. A estrutura dos próximos capítulos será organizada da seguinte forma: no Capítulo 2 apresenta trabalhos relacionados, os quais são abordados estudos prévios que fundamentam a pesquisa. No Capítulo 3 desenvolvido a fundamentação teórica, onde é explorado os conceitos que sustentam o sistema proposto. Já no Capítulo 4, há um detalhamento da arquitetura e outros requisitos do modelo proposto. Além disso, no Capítulo 5, capítulo que traz a metodologia utilizada nesse projeto, há a descrição dos procedimentos de coleta e análise de dados. A seção 6, apresenta e enfatiza os resultados obtidos, e ainda discute sobre esses dados, além de comparar com outros sistemas. E, por fim, o último Capítulo, que conclui o texto destacando as principais contribuições do estudo e sugerem direções para pesquisas futuras, enfatizando a relevância de sistemas de recomendação que melhorem a experiência do usuário.

2 Trabalhos Relacionados

Uluyagmur et al. (2012) desenvolveram um sistema de recomendação usando filtragem baseada em conteúdo, onde características dos filmes, como atores, diretores, gêneros e palavras-chave, são utilizadas para personalizar as recomendações ao perfil de cada usuário. Um exemplo prático desse tipo de abordagem pode ser visto na plataforma Rotten Tomatoes ¹, que analisa as características dos filmes para sugerir títulos que correspondam ao histórico de preferências do usuário.

No campo da filtragem colaborativa, (Sarwar et al. 2001) exploraram as limitações dessa técnica, destacando o problema do *"cold start"*, que ocorre ao tentar recomendar itens para novos usuários ou itens sem histórico. Essa abordagem é amplamente utilizada para criação de plataformas como o Letterboxd², onde as recomendações são geradas com base nas avaliações e listas criadas por outros usuários com interesses semelhantes. (Salloum e Rajamanthri 2021) também contribuíram para essa área com uma abordagem de filtragem colaborativa baseada em usuários, que identifica perfis semelhantes para oferecer recomendações mais precisas. Por sua vez, (Dwivedi e Islam 2023) aplicaram uma abordagem de filtragem colaborativa baseada em itens, onde as recomendações são baseadas na similaridade entre itens previamente avaliados positivamente, técnica que reforça a precisão das recomendações e é frequentemente utilizada em plataformas de recomendações sociais.

Por sua vez, os métodos de filtragem híbrida têm sido amplamente aplicados em plataformas de streaming como Netflix e Amazon Prime Video. Essas plataformas combinam filtragem colaborativa e baseada em conteúdo para melhorar a diversidade e a personalização das recomendações. Koren et al (2009) foram pioneiros na introdução de métodos híbridos mais complexos, incluindo redes neurais e análise de dados em larga escala. Esse avanço abriu caminho para modelos mais sofisticados que equilibram personalização e descoberta. Estudos mais recentes, como os de Alsekait (2024), desenvolveram sistemas híbridos com redes neurais profundas para capturar padrões complexos nas preferências dos usuários. De forma similar, (Setiawan e Arsyntania 2024) propuseram um sistema híbrido em cascata, combinando redes neurais convolucionais com técnicas colaborativas, garantindo recomendações diversificadas e relevantes para os usuários.

Essas diferentes abordagens de sistemas de recomendação, suas características e principais contribuições estão organizadas na Tabela 1 que destaca os trabalhos fundamentais e as contribuições mais recentes neste campo de estudo. Essa Tabela

¹ <https://www.rottentomatoes.com/about>

² <https://letterboxd.com/films/>

Oapresenta uma revisão detalhada das técnicas anteriores utilizadas, evidenciando os métodos de filtragem empregados, os autores responsáveis por cada avanço, os anos de publicação e exemplos práticos de plataformas que implementam essas tecnologias demonstrando avanços no uso de novas ferramnetas, como a utilização de redes neurais, e métodos combinados para aprimorar recomendações.

Tabela 1 – Comparação de métodos de recomendação

Nome do Autor	Ano de Publicação	Tipo de Filtragem (Exemplo de Plataforma)	Resumo da Obra
Uluayagmur	2012	Filtragem baseada em conteúdo (Rotten Tomatoes)	Sistema de recomendação utilizando características dos filmes para personalizar recomendações ao perfil do usuário.
Sarwar et al.	2001	Filtragem colaborativa - cold start (Letterboxd)	Exploraram as limitações da filtragem colaborativa, especialmente o problema de 'cold start' para novos usuários ou itens sem histórico.
Salloum e Rajamanthri	2021	Filtragem colaborativa baseada em usuários	Abordagem de filtragem colaborativa baseada em usuários, identificando perfis semelhantes para recomendações mais precisas.
Dwivedi e Islam	2023	Filtragem colaborativa baseada em itens (Plataformas de recomendações sociais)	Abordagem de filtragem colaborativa baseada em itens, recomendando com base na similaridade entre itens previamente avaliados positivamente.
Koren	2009	Filtragem híbrida (Netflix, Amazon Prime Video)	Introdução de métodos híbridos avançados, como redes neurais e análise em larga escala para equilibrar personalização e descoberta.
Alsekait	2021	Filtragem híbrida com redes neurais profundas	Sistema híbrido com redes neurais profundas para capturar padrões complexos nas preferências dos usuários.
Setiawan e Arsyantania	2024	Filtragem híbrida em cascata com redes neurais convolucionais	Sistema híbrido em cascata combinando redes neurais convolucionais e técnicas colaborativas para recomendações diversificadas.

Fonte: Elaborado pelo autor

3 Fundamentação Teórica

Para criar recomendações personalizadas, muitos sistemas utilizam diferentes métricas de similaridade, e diferentes tipos de filtragem que ajudam a medir de formas diversas a proximidade entre filmes com base em atributos compartilhados.

Nesta seção, são abordados três principais temas: o primeiro tema aborda a evolução dos sistemas de recomendação e detalha sobre os 3 tipos de filtragem (Filtragem Baseada em Conteúdo, Filtragem Colaborativa e Filtragem Híbrida), tipos essenciais para entendimento do projeto. Já o segundo tema aborda os principais fundamentos matemáticos e conceituais que sustentam o sistema de recomendação de filmes proposto. Em particular, exploram-se 3 métricas de similaridade aplicadas (Similaridade Cosseno, Correlação de Pearson e Índice de Jaccard), onde cada uma dessas possui uma abordagem específica para calcular a semelhança entre os itens ou perfis de usuários, além da clusterização que são utilizadas para calcular a afinidade entre filmes com características comuns, através de um sistema qualitativo ao invés de quantitativo. Já o último tema, descreve sobre os tipos de métricas utilizadas para avaliação do sistema. Para a avaliação, dividimos o processo em duas etapas: primeiramente, realizamos o cálculo da precisão do sistema, avaliando a proporção de recomendações corretas entre todas as recomendações feitas. Em seguida, a segunda parte da análise foca no recall, que mede a capacidade do sistema de recuperar itens relevantes para o usuário entre todos os itens relevantes disponíveis. Com estes dois dados, pode ser analisado o sistema e comparado com outros disponíveis.

No sistema proposto, as métricas de similaridade e a clusterização em conexão com diferentes tipos de filtragem fornecem opções independentes, como alternativas para personalizar a recomendação de filmes. Adicionalmente, uma métrica combinada utiliza os resultados de três dessas métricas de similaridade para gerar recomendações aprimoradas.

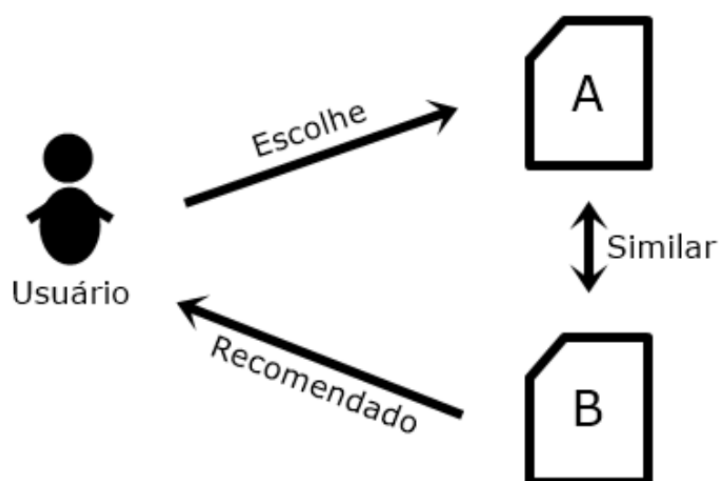
3.1 Evolução dos Sistemas de Recomendação de Filmes

Os sistemas de recomendação tornaram-se ferramentas cruciais na era digital, ajudando os usuários a encontrar conteúdos alinhados a seus interesses. A evolução dessas tecnologias é marcada por desafios que levaram à transição entre diferentes tipos de filtragem: começando pela filtragem baseada em conteúdo, passando pela filtragem colaborativa, e culminando em sistemas híbridos. Abaixo será detalhado cada tipo de filtragem e suas aplicações na área dos filmes.

3.1.1 Filtragem Baseada em Conteúdo

O conceito de filtragem baseada em conteúdo é amplamente utilizado em sistemas de recomendação de filmes. Esse tipo de filtragem analisa as características dos itens para fazer recomendações. No caso de filmes, isso inclui dados específicos, como gênero, diretor, elenco e sinopse. O sistema compara essas características com o histórico de preferências do usuário para sugerir filmes semelhantes aos que ele já apreciou. Por exemplo, se um usuário assistiu a filmes de ação com um determinado ator, o sistema recomendará outros filmes de ação com o mesmo ator ou de temática similar. A Figura 1 ilustra esse processo, onde o sistema analisa o objeto/filme A, procura um item similar (exemplificado pela letra B) e retorna essa recomendação ao usuário.

Figura 1 – Filtragem Baseada em conteúdo



Fonte: Adaptada de Rolim et al.(2017)

Contudo, foram identificados problemas no uso dessa técnica. Uma das limitações da filtragem baseada em conteúdo é a criação de uma "bolha de filtragem", onde o sistema expõe o usuário repetidamente a conteúdos semelhantes aos que ele já consumiu. Isso reduz a diversidade e cria um ciclo fechado, no qual o usuário interage com itens previsíveis, recebendo recomendações cada vez menos variadas. Esse processo pode limitar a descoberta de novos interesses e diminuir a satisfação ao longo do tempo, à medida que a experiência se torna repetitiva.

3.1.2 Filtragem Colaborativa

O conceito de filtragem colaborativa foi desenvolvido inicialmente para analisar o comportamento de visualização dos usuários e identificar padrões de consumo. A filtragem colaborativa pode ser dividida em duas abordagens principais: a baseada em usuários e a baseada em itens.

Uma das limitações comuns da filtragem colaborativa é o problema da "fria inicial" ou "*cold start*", que ocorre quando o sistema tem dificuldades para sugerir novos itens ou para usuários sem histórico de interação (Sarwar et al. 2001). Além disso, a filtragem colaborativa pode levar a um fenômeno conhecido como "conformidade", onde os usuários são expostos a recomendações populares, resultando em uma redução da diversidade nas sugestões. Esse cenário pode limitar a descoberta de novos conteúdos e reduzir a satisfação do usuário ao longo do tempo.

3.1.2.1 Filtragem Colaborativa Baseada em usuários

Na filtragem colaborativa baseada em usuários, o sistema recomenda filmes a um usuário com base nas avaliações de outros usuários com gostos semelhantes. Por exemplo, se o usuário A e o usuário B compartilham um histórico de avaliações parecido, o sistema pode sugerir a este último filmes que o usuário A gostou, mas que o usuário B ainda não assistiu. Essa abordagem se baseia na premissa de que, se dois usuários concordaram sobre um item no passado, é provável que também concordem sobre outros itens no futuro. Isso permite sugerir filmes bem avaliados por usuários com perfis semelhantes, identificado através de um padrão de consumo.

3.1.2.2 Filtragem Colaborativa Baseada em Itens

A filtragem colaborativa baseada em itens analisa as similaridades entre os itens (neste caso, filmes) com base nas avaliações dos usuários. O sistema identifica filmes que foram frequentemente avaliados da mesma forma e recomenda um filme a um usuário com base em itens que ele já avaliou positivamente. Por exemplo, se um filme A e um filme B receberam avaliações altas de um mesmo grupo de usuários, o sistema pode recomendar o filme B a quem gostou do filme A, ou seja, o sistema foca em recomendar filmes com características similares aos já assistidos pelo usuário, aprimorando a precisão das recomendações sem recorrer a abordagens híbridas.

3.1.3 Sistemas de Recomendação Híbridos

A ideia de sistemas híbridos, aplicados diretamente a recomendações de filmes, combina a filtragem colaborativa e baseada em conteúdo para oferecer recomendações mais diversificadas e precisas. Em sistemas híbridos, a filtragem colaborativa explora as preferências coletivas dos usuários, enquanto a filtragem baseada em conteúdo se concentra nas características dos itens, criando uma experiência personalizada e diversificada.

Esse tipo de sistema resolve desafios comuns em recomendações, como o problema de "*cold start*" e a "bolha de filtragem." Dessa forma, esses sistemas integram o histórico de visualizações e avaliações dos usuários com dados específicos dos

filmes, como gênero, elenco e diretores, oferecendo uma recomendação equilibrada que inclui tanto o que é popular quanto novos conteúdos alinhados ao perfil do usuário. Abordagens adicionais podem incorporar dados adicionais, como histórico de compras e interações em dispositivos conectados, para enriquecer as recomendações com uma camada de contexto adicional.

Apesar dos avanços, a filtragem híbrida também apresenta desafios, como a complexidade computacional e a necessidade de balanceamento cuidadoso para evitar recomendações irrelevantes. No entanto, pesquisas recentes confirmam a filtragem híbrida como uma das metodologias mais promissoras para oferecer recomendações diversificadas e satisfatórias em plataformas de filmes.

3.2 Métricas de Similaridade

As métricas de similaridade desempenham um papel crucial na recomendação de filmes, pois permitem quantificar a proximidade entre diferentes itens com base em suas características. As métricas escolhidas neste projeto, como mencionado anteriormente (similaridade cosseno, correlação de Pearson e índice de Jaccard) são utilizadas para calcular a afinidade entre filmes e, assim, recomendar aqueles com maior semelhança.

3.2.1 Similaridade Cosseno

A similaridade cosseno (Laboratório de Aplicações de Machine Learning em Finanças e Organizações 2018) é uma medida de similaridade vetorial que calcula o cosseno do ângulo entre dois vetores em um espaço multidimensional. Essa métrica é amplamente utilizada em sistemas de recomendação para avaliar a proximidade entre itens ou perfis de usuários. Formalmente, para dois vetores A e B, a similaridade cosseno é dada pela fórmula apresentada na Equação 3.1:

$$\text{similarity}(A, B) = \cos(A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \quad (3.1)$$

- $A \cdot B$ representa o produto escalar dos vetores A e B, que é a soma dos produtos dos valores correspondentes em cada vetor.
- $\|A\|$ e $\|B\|$ representam os comprimentos dos vetores A e B, calculadas como a raiz quadrada da soma dos quadrados de seus elementos.
- O resultado da similaridade cosseno varia entre -1 e 1, onde um valor de 1 indica que os vetores são exatamente iguais em direção (máxima similaridade), 0 indica que são ortogonais (sem similaridade) e -1 indica que os vetores são opostos.

No contexto de recomendação de filmes, cada vetor representa as características de um filme, e a similaridade cosseno mede o grau de alinhamento entre essas características, resultando em valores entre 0 e 1, onde valores mais distantes de 1 são menos semelhantes ao filme.

3.2.2 Correlação de Pearson

A correlação de Pearson (Pearson 1895) é uma métrica que quantifica a relação linear entre duas variáveis, permitindo identificar padrões de comportamento e preferências dos usuários. É expressa pela fórmula da Equação 3.2:

$$P = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2} \cdot \sqrt{\sum (y_i - \bar{y})^2}} \quad (3.2)$$

Onde:

- x_i e y_i representam os valores individuais das variáveis X e Y .
- \bar{x} e \bar{y} são as médias de X e Y , respectivamente.
- $\sum ((x_i - \bar{x})(y_i - \bar{y}))$ calcula o somatório do produto das diferenças entre cada valor e a média das variáveis.
- $\sqrt{\sum (x_i - \bar{x})^2}$ e $\sqrt{\sum (y_i - \bar{y})^2}$ calculam as raízes quadradas do somatório das diferenças ao quadrado de cada valor e a média, correspondendo às variâncias das variáveis.
- Essa fórmula resulta em um valor entre -1 e 1, onde 1 indica uma correlação positiva perfeita, -1 uma correlação negativa perfeita e 0 indica que não há correlação linear entre as variáveis.

Assim como a anterior, na recomendação de filmes, a correlação de Pearson é útil para comparar as preferências dos usuários em relação a um mesmo conjunto de filmes, permitindo que o sistema identifique tendências de avaliação similares.

3.2.3 Índice de Jaccard

O índice de Jaccard (Jaccard 1901) é uma métrica de similaridade que mede a intersecção entre dois conjuntos dividida pela união desses conjuntos, sendo particularmente útil para dados categóricos. Abaixo há a fórmula desse índice indicado pela Equação 3.3 além de uma explicação sobre ela.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (3.3)$$

Explicando um pouco a Equação (3.3):

- $|A \cap B|$ representa o número de elementos em comum entre os conjuntos A e B (interseção).
- $|A \cup B|$ representa o número total de elementos na união dos conjuntos A e B .
- O índice de Jaccard varia entre 0 e 1, onde 1 indica que os conjuntos são idênticos (máxima similaridade), e 0 indica que não possuem elementos em comum.

No contexto de recomendação de filmes, essa métrica é útil para medir a similaridade entre filmes com características compartilhadas, como gêneros ou atores, permitindo recomendar filmes que possuam uma maior quantidade de características em comum.

3.2.4 Métrica Combinada

Para obter recomendações ainda mais precisas, o sistema também oferece uma métrica combinada que integra os resultados das três métricas anteriores (similaridade cosseno, correlação de Pearson e índice de Jaccard). Essa abordagem visa maximizar a precisão da recomendação ao combinar diferentes perspectivas de similaridade em uma única medida agregada. A combinação dessas métricas é realizada por meio de uma média ponderada utilizando valores distintos para cada, resultando em uma seleção dos filmes que apresentam maior afinidade global.

O trabalho em si aplica essas métricas de forma independente, e o usuário pode optar por qualquer uma delas ou pela métrica combinada para gerar recomendações com diferentes perspectivas de similaridade entre filmes.

3.3 Clusterização

A clusterização é um dos métodos utilizado em análise de dados, que visa agrupar itens ou objetos com características semelhantes, facilitando a identificação de padrões e relações ocultas nos dados. Ao contrário das métricas de similaridade, que quantificam a afinidade entre pares de itens, a clusterização permite organizar um conjunto de dados em grupos naturais, onde os elementos de cada grupo compartilham propriedades em comum. Essa abordagem de separação em grupos, ilustrada na Figura 2, é particularmente útil em sistemas de recomendação, onde o objetivo é apresentar opções relevantes ao usuário. A seguir, após a ilustração, será discutido o DBSCAN, uma técnica de clusterização baseada em densidade que se destaca pela sua capacidade de identificar clusters de forma arbitrária e lidar com dados ruidosos.

Figura 2 – Clusterização



Fonte: Adaptada de (Gomes 2024)

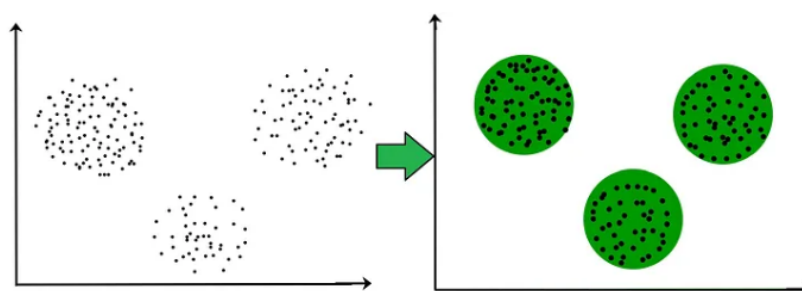
3.3.1 DBSCAN

O DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*) é um algoritmo de clusterização que agrupa pontos em regiões densamente povoadas, enquanto identifica pontos isolados como ruídos. O algoritmo opera em duas etapas principais:

- Definição dos parâmetros de entrada, que incluem o raio de busca (ϵ) e o número mínimo de pontos (MinPts) necessários para formar um cluster.
- Envolvimento da identificação de pontos centrais que possuem pelo menos MinPts vizinhos dentro do raio ϵ .

Os pontos centrais são então agrupados, e a expansão do *cluster* continua até que não haja mais pontos densamente conectados. Essa abordagem (exemplificada na Figura 3) permite que o DBSCAN encontre clusters de formatos variados e tamanhos diferentes, proporcionando uma alternativa robusta às técnicas de clusterização tradicionais que requerem a definição prévia do número de *clusters*.

Figura 3 – Funcionamento do DBSCAN



Fonte: Adaptada de (Monteiro e Carl 2020)

3.4 Métricas de avaliação

A avaliação de sistemas de recomendação depende de métricas que refletem a qualidade das sugestões geradas e sua adequação aos interesses dos usuários. Entre as métricas essenciais estão a precisão e o recall, que abordam diferentes aspectos do desempenho do sistema: enquanto a precisão mede o quão assertivo o sistema é ao recomendar itens relevantes, o recall avalia sua abrangência em identificar opções de interesse. Essas métricas permitem não apenas verificar a eficácia do sistema, mas também guiar melhorias e comparações com outras abordagens.

3.4.1 Precisão: Definição e Cálculo

Precisão é uma métrica que mede a proporção de recomendações relevantes entre todas as recomendações feitas. Em outras palavras, indica quão bem o sistema está em acertar ao recomendar filmes que realmente são do interesse do usuário. A equação 3.4 mostra qual é a fórmula para precisão:

$$\text{Precisão} = \frac{TP}{TP + FP} \quad (3.4)$$

Onde:

- *TP* (Verdadeiros Positivos) são os filmes que foram recomendados e que realmente são relevantes.
- *FP* (Falsos Positivos) são os filmes que foram recomendados, mas que não são relevantes.
- Valores de precisão próximos de 1 indicam que uma alta proporção das recomendações feitas pelo sistema são relevantes, o que é desejável em um sistema de recomendação (Jannach e Adomavicius 2016).

3.4.2 Recall: Definição e Cálculo

Recall, por outro lado, mede a capacidade do sistema de recomendar filmes relevantes dentre todos os filmes que poderiam ser recomendados. Ele reflete a abrangência do sistema em captar opções que atendem ao gosto do usuário. O recall, representado pela equação 3.5 é calculado através da formula abaixo:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3.5)$$

Onde:

- TP Continua sendo os Verdadeiros Positivos.
- FN (Falsos Negativos) são os filmes relevantes que não foram recomendados pelo sistema.
- Valores mais altos de recall indicam que o sistema é eficaz em capturar uma grande variedade de itens de interesse, proporcionando uma experiência mais abrangente (Uluyagmur 2012).

Concluindo, a fundamentação teórica apresentada abrange os métodos utilizados para a recomendação de filmes no sistema proposto. Cada métrica contribui para a identificação de filmes com características ou preferências semelhantes, oferecendo ao usuário uma variedade de critérios para personalizar suas recomendações. A utilização da clusterização complementa os métodos para obtenção da similaridade ao agrupar filmes com base em características comuns, proporcionando uma abordagem diversa para explorar o catálogo. No próximo capítulo, o modelo proposto será detalhado, abordando as etapas práticas e as decisões de implementação que sustentam o desenvolvimento do sistema de recomendação de filmes.

4 Modelo Proposto

O sistema de recomendação de filmes desenvolvido neste trabalho, chamado FilmMatch, foi criado para sugerir filmes parecidos com o filme utilizado como referência pelo usuário. O modelo usa várias métricas de similaridade aplicadas a características específicas dos filmes, permitindo que o usuário selecione a métrica desejada. Analisando a aplicação, O FilmMatch utiliza separadamente, métodos de filtragem baseada em conteúdo e filtragem colaborativa para identificar filmes com características e padrões de avaliação semelhantes. Além disso, oferece uma opção de filtragem híbrida que combina essas abordagens para equilibrar melhor as recomendações.

4.1 Filtragem Baseada em Conteúdo

A filtragem baseada em conteúdo do FilmMatch considera atributos específicos dos filmes, como gênero, elenco e diretores, para calcular a similaridade entre eles. Os métodos usados para esse tipo de filtragem são descritos a seguir:

4.1.1 Índice de Jaccard

O Índice de Jaccard é utilizado para medir a sobreposição entre conjuntos de atributos, como gêneros e elenco dos filmes. No FilmMatch, essa métrica é calculada transformando os atributos dos filmes, como ator e gênero, em variáveis binárias, onde o valor 1 representa a presença de um determinado gênero ou ator e o valor 0 representa a ausência desses atributos. Essa abordagem permite medir a similaridade entre o filme selecionado e os outros filmes, analisando o grau de sobreposição nos atributos de conteúdo.

4.1.2 Clusterização por Gênero e Diretores

A métrica de Clusterização por Gênero e Diretores agrupa filmes com base nas características de gênero e diretores baseando-se na técnica de DBSCAN. Esse tipo de clusterização permite que filmes com perfis semelhantes sejam agrupados, aumentando a chance de serem recomendados uns aos outros. Ao identificar grupos de filmes com características de gênero e diretores próximos, a clusterização ajuda a melhorar a precisão das recomendações, especialmente ao segmentar filmes com temas e estilos semelhantes.

4.2 Filtragem Colaborativa

Além das características de conteúdo, o FilmMatch utiliza as avaliações dos usuários de duas grandes plataformas de avaliação, o IMDb (*Internet Movie Database*) e TMDb (*The Movie Database*) para identificar filmes com padrões de avaliação semelhantes.

4.2.1 Correlação de Pearson

A Correlação de Pearson mede a relação linear entre as avaliações dos filmes, capturando tendências de popularidade e padrões de avaliação, apenas as avaliações das duas plataformas. Com essas variáveis, a métrica consegue indicar quais são os filmes com melhor índice de satisfação do usuário.

4.2.2 Similaridade Cosseno

Já na Similaridade Cosseno além de utilizar as notas, também é aplicada o uso de variáveis contínuas (como duração e contagem de votos), ajudando a identificar filmes que têm popularidade ou características numéricas semelhantes.

4.2.3 Filtragem Híbrida (Similaridade Total)

O modelo também oferece uma opção de filtragem híbrida chamada Similaridade Total, que combina 60 por cento do Índice de Jaccard (baseado em conteúdo), 20 por cento da Similaridade Cosseno e 20 por cento da Correlação de Pearson (ambas colaborativas).

Para escolha desses valores, Foram realizados testes utilizando cada uma das métricas de similaridade implementadas no sistema, revelando diferenças significativas nos resultados. O índice de Jaccard apresentou, na maioria dos testes, uma pontuação média em torno de 0.4, enquanto as métricas de Correlação de Pearson e Similaridade Cosseno alcançaram resultados superiores, próximos de 0.8 e 0.9, respectivamente. Esses resultados evidenciam que diferentes métricas capturam aspectos distintos da similaridade entre os filmes, sendo necessário combinar suas forças para alcançar um sistema de recomendação mais robusto e eficiente.

A escolha da maior parte para o Índice de Jaccard reflete a ênfase nas características de conteúdo, como gênero e elenco, uma vez que esses elementos são essenciais para captar a similaridade temática e estilística entre os filmes. Já a atribuição das demais porcentagens Similaridade Cosseno e a Correlação de Pearson permite considerar as avaliações dos usuários, garantindo que filmes com popularidade

e padrões de avaliação semelhantes também tenham relevância nas recomendações, sem sobrepor-se às características de conteúdo.

Com isso, o sistema consegue equilibrar as características de conteúdo com as avaliações dos usuários, gerando uma recomendação mais abrangente.

4.2.4 Funcionamento do Modelo

O funcionamento do FilmMatch envolve as seguintes etapas:

- **Escolha do Filme e da Métrica:** O usuário escolhe um filme de referência e uma métrica de similaridade para iniciar o processo de recomendação.
- **Cálculo de Similaridade::** O modelo aplica a métrica selecionada para calcular a similaridade entre o filme escolhido e os outros filmes do conjunto de dados. Dependendo da métrica, o cálculo pode ser baseado nas características de determinado filme (como gênero, elenco e diretores), nas avaliações e padrões de popularidade, ou em uma combinação híbrida.
- **Classificação e Exibição de Recomendações::** O modelo classifica os filmes de acordo com a pontuação de similaridade e exibe os cinco filmes mais similares ao selecionado pelo usuário. Esse processo de recomendação permite uma experiência interativa, já que o usuário pode selecionar a abordagem que deseja para obter sugestões de filmes.

Dessa forma, o modelo FilmMatch utiliza abordagens dos 3 tipos de filtragem permitindo flexibilidade na escolha dos atributos de similaridade e oferecendo uma experiência personalizada ao usuário.

5 Metodologia

5.1 Ambiente e Ferramentas

O sistema FilmMatch foi desenvolvido no Google Colab, utilizando a linguagem de programação Python e um conjunto de bibliotecas especializadas para manipulação de dados e criação de uma interface interativa.

5.1.1 Google Colab

O Google Colab é uma plataforma gratuita de notebooks baseada em nuvem, oferecida pelo Google, que permite o desenvolvimento de projetos em Python. Uma das principais vantagens desse ambiente é a possibilidade de acesso a GPUs e TPUs gratuitas, o que facilita o processamento de grandes volumes de dados e acelera as tarefas de machine learning. Além disso, o Colab integra-se facilmente com o Google Drive, possibilitando que conjuntos de dados sejam carregados, armazenados e acessados diretamente do Drive. Essa integração foi crucial para facilitar o armazenamento dos três conjuntos de dados utilizados, que foram carregados e manipulados diretamente no notebook do Colab, garantindo agilidade no desenvolvimento e processamento.

5.1.2 Python

A linguagem de programação Python foi escolhida devido à sua simplicidade e facilidade de aprendizado, além da grande quantidade de bibliotecas especializadas e ampla adoção na área de ciência de dados e machine learning. A flexibilidade do Python foi essencial para manipular dados, criar funções customizadas para cálculos de similaridade e integrar o sistema a uma interface interativa.

5.1.2.1 Bibliotecas utilizadas

Abaixo há uma apresentação de todas as bibliotecas utilizadas no ambiente, além de seus devidos usos na implementação do sistema.

5.1.2.1.1 Pandas

A biblioteca Pandas foi fundamental para manipulação e organização dos dados, permitindo a importação, limpeza e transformação dos dados de forma eficiente. Com pandas, foi possível realizar o pré-processamento e a junção dos três conjuntos de

dados em um único DataFrame. além disso com esta biblioteca foi possível realizar o One-Hot Encoding para as colunas de diretores, gêneros e atores.

5.1.2.1.2 Numpy

A biblioteca numpy foi utilizada para realizar operações numéricas e cálculos vetorizados. Ela facilitou o processamento das variáveis contínuas e transformações nos dados, que foram preparadas para cálculos de similaridade e operações de clusterização.

5.1.2.1.3 Scikit-Learn

A scikit-learn foi utilizada para implementar métricas de similaridade (como a Similaridade Cosseno e a Correlação de Pearson) e o algoritmo de clusterização DBSCAN. A função "cosine_similarity" permitiu a comparação de variáveis contínuas após a padronização com StandardScaler, enquanto o DBSCAN foi aplicado para agrupar filmes com gêneros e diretores semelhantes.

5.1.2.1.4 Ipywidgets

A biblioteca ipywidgets permitiu a criação de uma interface interativa no Google Colab. Com ipywidgets, elementos como menus suspensos e botões foram implementados, possibilitando ao usuário selecionar o filme e a métrica de similaridade desejada de forma intuitiva. A exibição dos filmes recomendados, com títulos e pôsteres, melhora a experiência do usuário, tornando o sistema mais amigável e visual. Abaixo, na Figura 4, é apresentada a parte gráfica dessa aplicação, mostrando a tela inicial, antes do cálculo das recomendações. Nesta tela é possível observar que há um botão para seleção do filme e logo abaixo a descrição do filme selecionado. Pouco abaixo dessa descrição, há mais dois botões, um para seleção da metodologia e outro para calcular essa recomendação. Além disso vale ressaltar que em cada um desses botões, como exemplificado na figura 5 (na escolha a metodologia), possui uma quantidade de opções para que o software recomende de maneira correta e diversa, dependendo apenas do filme e métrica escolhida.


5.2 Datasets

Foi utilizado para criação do sistema, três datasets ((Netflix Movies and TV Shows, IMDB Movies Dataset e TMDb Movies (900k movies + daily updates), ambos retirados da plataforma Kaggle. Dentro de cada dataset cada linha representa uma observação, e cada coluna, uma variável ou característica dos dados. Explicando

Figura 4 – Plataforma FilmMatch

Selecione um filme da lista abaixo:

Filme:



Diretor: Grant Sputore
Ano de Lançamento: 2019
Duração: 114
Gênero: Drama, Mystery, Sci-Fi
Descrição:
 Following humanity's mass extinction, a teen raised alone by a maternal droid finds her entire world shaken when she encounters another human.

Escolha a metodologia:


Método:

Fonte: Elaborado pelo autor

Figura 5 – Painel de opções das métricas do FilmMatch

Selecione um filme da lista abaixo:

Filme:



Diretor: Raja Gosnell
Ano de Lançamento: 2011
Duração: 103
Gênero: Animation, Adventure, Comedy
Descrição:
 When evil Gargamel tries to capture them, the Smurfs flee their woodland home, pass through a magic portal and find themselves stranded in New York.

Escolha a metodologia:

Método:

- Jaccard
- Cosseno
- Pearson**
- Total
- Clusterização

Fonte: Elaborado pelo autor

um pouco sobre o Kaggle, ele é uma plataforma online que oferece um ambiente colaborativo para cientistas de dados e entusiastas de machine learning compartilharem e aprimorarem suas habilidades. É utilizado também para explorar e analisar dados, competir em desafios de ciência de dados e machine learning, e também para acessar conjuntos de dados públicos. A seguir será descrito um pouco de cada dataset, além de explicar o motivo que foram escolhidos.

5.2.1 Netflix Movies and TV Shows

O dataset "Netflix Movies and TV Shows", fornece informações detalhadas sobre o catálogo de conteúdo da plataforma, incluindo 15 atributos diferentes como o nome do título (*title*), diretor (*director*), ano de lançamento (*release_year*), duração (*duration*),

elenco (*cast*), e uma breve descrição (*description*), porém apenas os exemplificados foram utilizados. Esses dados foram fundamentais para o desenvolvimento do sistema de recomendação, permitindo aumentar a diversidade de análises de similaridade entre títulos com base diferentes tipos de métricas, e agrupamento de características para serem mostradas na tela final como ano de lançamento e descrição.

5.2.2 IMDB Movies Dataset

O IMDB Movies Dataset, criado por Aman Barthwa, foi outra fonte essencial de dados para aprimorar o sistema de recomendação, oferecendo informações como o pôster do filme (*Poster*), título (*Title*), gênero (*Genre*), classificação (*Rating*) e número de votos (*Votes*). O pôster foi integrado ao sistema para exibir a imagem do filme selecionado e do recomendado. O título e o gênero possibilitaram a integração entre diferentes datasets do sistema de recomendação (utilizados amplamente para verificação) enquanto a classificação e o número de votos foram empregados na filtragem colaborativa, permitindo uma seleção mais refinada de filmes com base em avaliações e popularidade, ou seja, esses atributos contribuíram para a parte gráfica, mas também para a parte funcional do sistema.

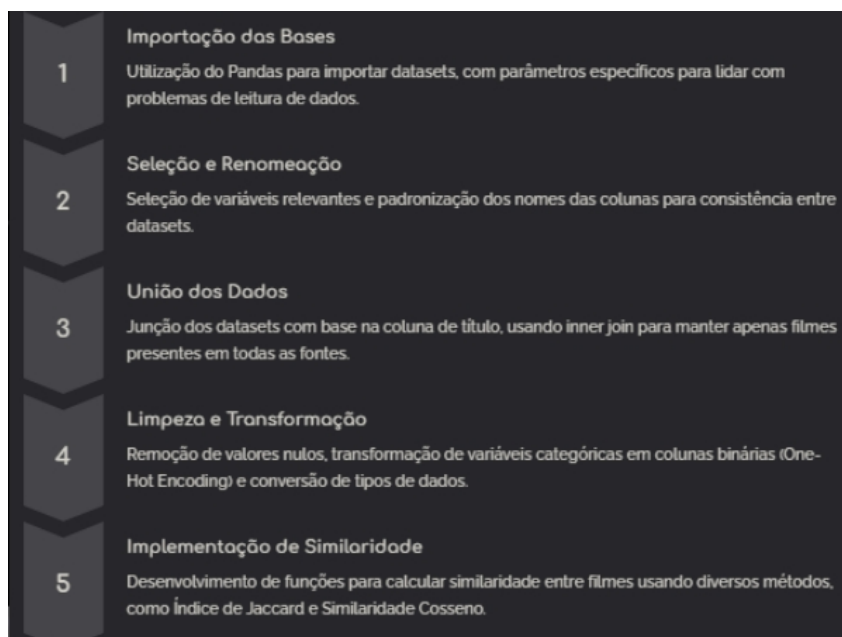
5.2.3 TMDb Movies (900k movies + daily updates)

O dataset TMDb Movies (900k movies + daily updates), criado por Alan Vourc'h, proveniente do The Movie Database (TMDb), oferece uma base extensa e frequentemente atualizada de filmes, contendo mais de 900 mil títulos (contando as duplicações e espaços com erros no preenchimento). Esse dataset é especialmente valioso por incluir avaliações detalhadas dos usuários, o que enriquece a análise de preferências no sistema de recomendação. No projeto, foram utilizadas apenas colunas de título (*title*), média de votos (**vote_average**) e contagem de votos (*vote_count*). Essas informações fornecem uma perspectiva adicional de avaliação, complementando a filtragem colaborativa e permitindo uma recomendação mais variada e alinhada aos interesses dos usuários.

5.3 Etapas de Processamento

Para garantir que o sistema de recomendação funcione de maneira eficaz, foi necessário seguir uma série de etapas de processamento dos dados, desde a seleção de variáveis relevantes até a implementação das funções de similaridade. Essas etapas, ilustradas na Figura 6, permitiram estruturar e preparar os dados de diferentes datasets, adequando-os aos requisitos específicos das análises e algoritmos de recomendação. As etapas estão divididas em cinco fases principais, detalhadas a seguir:

Figura 6 – Etapas de Processamento dos dados



Fonte: Elaborado pelo autor

5.3.1 Importação das bases

Utilizando Python e a biblioteca pandas, os datasets foram importados com a função `pd.read_csv`, configurada com parâmetros específicos para lidar com problemas comuns de leitura de dados. A Figura 7 reflete o código a ser mencionado para importação das bases.

Figura 7 – Código de importação das bases

```
# importar bibliotecas do python e arquivos da pasta do google drives
import os
from google.colab import drive
drive.mount('/content/drive', force_remount = True)

# caminhos para cada arquivo da pasta (primeiro o do titulo, segundo imagem e terceiro classificação)
titles_file_path = '/content/drive/MyDrive/Arquivos filmes netflix/netflix_titulos_filmes.csv'
images_file_path = '/content/drive/MyDrive/Arquivos filmes netflix/imdb_movies_dataset.csv'
ratings_file_path = '/content/drive/MyDrive/Arquivos filmes netflix/TMDB_all_movies_notas.csv'

# importar bibliotecas pandas e numpy
import pandas as pd
import numpy as np

# importar bases tendo em vista a coluna como referência
titles_df = pd.read_csv(titles_file_path, encoding='ISO-8859-1', sep = ';') # Try semicolon
images_df = pd.read_csv(images_file_path, encoding='ISO-8859-1', sep = ';') # Try semicolon

# Pular linhas da base da avaliação caso haja problemas
ratings_df = pd.read_csv(ratings_file_path, encoding='ISO-8859-1', sep = ';', on_bad_lines='skip')
# se voce quiser ver quais linhas foram puladas:
ratings_df = pd.read_csv(ratings_file_path, encoding='ISO-8859-1', sep = ';', on_bad_lines='warn')
```

Fonte: Elaborado pelo autor

O parâmetro `encoding='ISO-8859-1'` foi empregado para definir a codificação correta do arquivo, essencial para evitar erros de leitura de caracteres especiais. Já o parâmetro `sep=';'` especifica o delimitador utilizado nos arquivos, garantindo que o pandas interprete corretamente as colunas. Além destes, o argumento `on_bad_lines='skip'`

permite que linhas com problemas sejam ignoradas durante a leitura. Esses parâmetros em conjunto, foram usados para garantir uma leitura correta dos dados.

5.3.2 Seleção e Renomeação das Variáveis

As variáveis de interesse foram selecionadas em cada dataset para reduzir a quantidade de dados e focar apenas nos atributos necessários ao sistema de recomendação. Para garantir consistência e facilitar a integração, as colunas de cada dataset foram renomeadas. Após a seleção e padronização das colunas, feitas com o código da Figura 8, os datasets estavam prontos para a próxima etapa de integração.

Figura 8 – Código de Seleção das variáveis e renomeação de parte das bases

```
import pandas as pd

# Selecionar colunas relevantes de cada DataFrame
titles_selected = titles_df[['title', 'director', 'release_year', 'duration', 'description', 'cast']]
images_selected = images_df[['Poster', 'Title', 'Genre', 'Rating', 'Votes']]
ratings_selected = ratings_df[['title', 'vote_average', 'vote_count']]

# Renomear colunas para facilitar a junção (deixar todas minúsculas e iguais aos outros)
images_selected = images_selected.rename(columns={'Title': 'title'})
images_selected = images_selected.rename(columns={'Poster': 'poster'})
images_selected = images_selected.rename(columns={'Genre': 'genre'})
images_selected = images_selected.rename(columns={'Rating': 'rating_imdb'})
images_selected = images_selected.rename(columns={'Votes': 'count_votes_imdb'})
ratings_selected = ratings_selected.rename(columns={'vote_average': 'rating_tmdb'})
ratings_selected = ratings_selected.rename(columns={'vote_count': 'count_votes_tmdb'})
```

Fonte: Elaborado pelo autor

5.3.3 União dos Dados

Os datasets foram unidos com base na coluna title, usando junções internas (inner join) para garantir que apenas os filmes presentes em todas as fontes fossem mantidos. Explicando um pouco do código, representado na Figura 9, o código utiliza o campo título e o compara com o título da outra tabela. Caso for igual, elas se juntam e se for diferente aquele dado não é colocado na nova tabela. Duplicatas e valores nulos foram removidos para melhorar a qualidade dos dados.

Figura 9 – Código de União das bases

```
# Mesclar os DataFrames usando a coluna 'título' como chave
merged_df = pd.merge(titles_selected, images_selected, left_on='title', right_on='title', how='inner')
merged_df = pd.merge(merged_df, ratings_selected, left_on='title', right_on='title', how='inner')

# Caso queira ver quantos filmes tinha antes da remoção de duplicados e linhas com algum valor nulo:
num_filmes_antes_remocao = len(merged_df)
#print(f"Antes da remoção de linhas com valor nulo e duplicados: {num_filmes_antes_remocao}")

# Remover linhas com qualquer valor nulo
merged_df = merged_df.dropna()

# Ordenar pelo número de votos em ordem decrescente e remover duplicatas mantendo o com maior número de votos
merged_df = merged_df.sort_values(by='count_votes_imdb', ascending=False).drop_duplicates(subset='title')
```

Fonte: Elaborado pelo autor

5.3.4 Limpeza e Transformação

Na etapa de limpeza e transformação dos dados, foram realizadas operações essenciais para garantir que o DataFrame estivesse preparado para as análises de similaridade. Abaixo, são descritas as etapas de remoção de valores nulos, transformação de variáveis categóricas em colunas binárias (One-Hot Encoding), e conversão de variáveis de contagem de votos de float para int.

5.3.4.1 Remoção de Valores Nulos e Zerados

Para assegurar que o conjunto de dados fosse confiável, foi necessário remover registros com valores nulos ou zerados em colunas importantes para a recomendação. Dados ausentes ou valores zerados podem indicar informações incompletas ou inconsistentes, o que prejudica a eficácia das análises de similaridade, pois esses valores podem distorcer as métricas. A remoção desses valores é um passo essencial para garantir a integridade e qualidade dos dados utilizados pelo sistema.

5.3.4.2 Transformação das Variáveis Categóricas em Colunas Binárias (One-Hot Encoding)

As variáveis categóricas, como diretores, elenco e gêneros, foram transformadas em colunas binárias usando One-Hot Encoding. Essa técnica cria uma nova coluna para cada categoria, onde o valor é 1 se o filme pertence àquela categoria e 0 caso contrário. Esse processo é crucial para análise de similaridade, pois converte categorias textuais em uma representação numérica, facilitando o cálculo de distâncias entre filmes. O One-Hot Encoding permite que o sistema reconheça características como a presença de diretores ou gêneros específicos, que são fundamentais para personalizar as recomendações. Abaixo há um código que exemplifica uma das transformações feitas (transformação de diretores nessas colunas binárias).

5.3.4.3 Conversão de Variáveis de Contagem de Votos de Float para Int das Variáveis Categóricas em Colunas Binárias (One-Hot Encoding)

As colunas de contagem de votos, que originalmente estavam no formato float, foram convertidas para o tipo int. A conversão de float para int é vantajosa em variáveis de contagem, pois elimina casas decimais que não são relevantes para o número de votos. Isso simplifica a manipulação e reduz o uso de memória, tornando o DataFrame mais leve e as operações matemáticas mais eficientes.

5.3.5 Implementação das Funções de Similaridade

Cinco funções de similaridade foram implementadas, conectadas à interface gráfica para o cálculo das recomendações:

- Índice de Jaccard;
- Similaridade Cosseno;
- Correlação de Pearson;
- Clusterização com DBSCAN; ;
- Similaridade Total.

Cada uma permite um tipo de recomendação com base nas características de conteúdo e avaliações dos filmes. Na sequência, será descrito cada uma das implementações de funções.

5.3.5.1 Índice de Jaccard

A implementação do Índice de Jaccard em Python começa com a função `indice_jaccard`, que recebe dois filmes como entrada e extrai seus gêneros e elenco, convertendo essas informações em conjuntos. A função calcula a interseção e a união desses conjuntos para determinar a similaridade entre eles, retornando a média das similaridades de gênero e elenco. A função `calcular_jaccard` então aplica o índice de Jaccard entre o filme selecionado e todos os outros filmes do dataset, ordenando as recomendações com base na similaridade em ordem decrescente. O código em Python permite uma recomendação eficiente com base nas características temáticas dos filmes

5.3.5.2 Similaridade Cosseno

A implementação em Python utiliza a função `cosine_similarity` da biblioteca `sklearn.metrics.pairwise`. Primeiramente, as colunas numéricas relevantes, como `rating_imdb`, `rating_tmdb`, `duration`, entre outras, são padronizadas com `StandardScaler`. A função `calcular_cosseno` calcula a similaridade do cosseno entre o filme selecionado e todos os outros filmes, gerando uma lista ordenada de títulos com base na similaridade. A abordagem em Python permite uma comparação eficiente de vetores de características numéricas entre filmes, facilitando a recomendação com base em dados quantitativos.

5.3.5.3 Correlação de Pearson

Em Python, a função `calcular_pearson` utiliza `pearsonr` da biblioteca `scipy.stats` para calcular a correlação de Pearson entre o filme selecionado e os demais filmes do dataset. As variáveis numéricas são previamente padronizadas para garantir a consistência dos cálculos. A função retorna uma lista de filmes ordenada por correlação, facilitando a recomendação com base na relação linear entre atributos. A implementação em Python permite calcular rapidamente a similaridade entre os filmes com base em correlações estatísticas.

5.3.5.4 DBSCAN

A implementação utiliza o algoritmo DBSCAN da biblioteca `sklearn.cluster`. O código combina variáveis indicadoras de gênero e diretor em uma matriz, que é processada pelo DBSCAN para identificar clusters densos. A função `clusterizacao` aplica o DBSCAN e armazena o cluster de cada filme, permitindo que a recomendação sugira filmes do mesmo grupo. Em Python, essa abordagem facilita a recomendação de filmes com características estruturais comuns, especialmente em grandes bases de dados.

5.3.5.5 Similaridade Total

A implementação em Python começa com a função `calcular_similaridade_total`, que utiliza as funções de Jaccard, Cosseno e Pearson, aplicando pesos específicos para cada uma (0.6 para Jaccard, 0.2 para Cosseno e 0.2 para Pearson). O código calcula uma pontuação ponderada para cada filme, somando as similaridades ajustadas por peso e ordenando os filmes com base na similaridade total. A função resulta em uma recomendação que captura de forma abrangente as características dos filmes.

6 Resultados

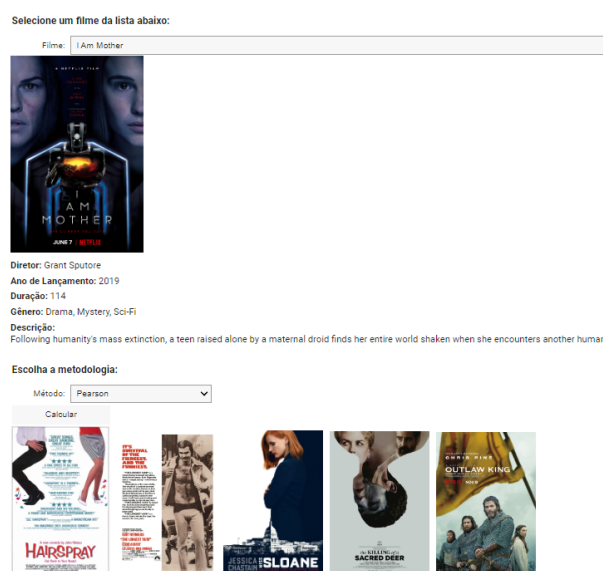
Neste capítulo, serão apresentados os resultados obtidos com o sistema de recomendação de filmes desenvolvido. Além disso, será abordado o tempo de execução do sistema, proporcionando uma visão da eficiência do modelo. Por fim, serão realizadas avaliações e comparações dos resultados avaliativos com outros sistemas de recomendação já existentes na literatura, destacando as forças e fraquezas do nosso sistema em relação a eles.

6.1 Recomendações de filmes

O sistema foi projetado para gerar recomendações com base em um filme selecionado pelo usuário, além da métrica que auxilia nessa recomendação. Para cada filme escolhido, diante de diferentes métricas, são recomendados algumas obras diferentes. Abaixo na figura 10 e 11, são apresentados respectivamente filmes recomendados utilizando a correlação de pearson e a similaridade total

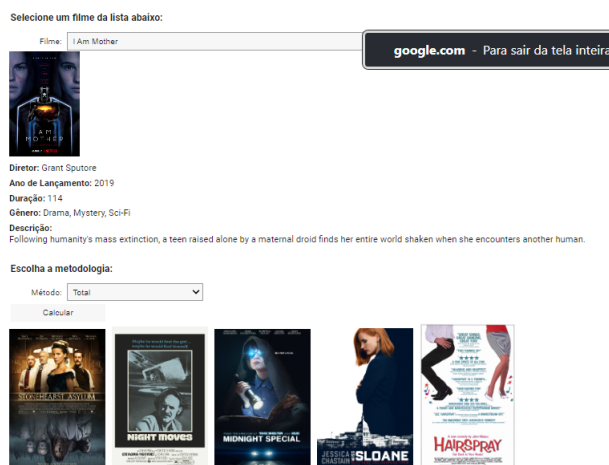
Como ambos possuem algumas métricas em comum, parte dos filmes, acabam aparecendo em ambos, conforme ilustrado na comparação entre a Figura 6 e a 7.

Figura 10 – Top 5 - correlação de pearson



Fonte: Elaborado pelo autor

Figura 11 – Top 5 - similaridade total



Fonte: Elaborado pelo autor

6.2 Aproximação e Tempo Real do Tempo de Execução por Métrica

6.2.1 Índice de Jaccard

O cálculo do Índice de Jaccard envolve a comparação de cada filme com todos os outros, resultando em uma complexidade de $O(n^2)$, onde n é o número de filmes. Para um conjunto de dados com 1000 filmes, essa métrica pode levar de 2 a 5 segundos para calcular, dependendo da implementação e da eficiência das operações de conjunto. Olhando na prática, o resultado desse sistema foi de 2.3 segundos, comprovando que estava correto o tempo de inicialização e execução da métrica

6.2.2 Similaridade Cosseno

O cálculo da Similaridade Cosseno é mais eficiente, pois utiliza operações vetoriais, resultando em uma complexidade de $O(n)$. Para 1000 filmes, essa métrica pode levar em torno de 0.5 a 1 segundo para calcular, proporcionando uma execução mais rápida em comparação com a métrica de Jaccard. Na execução, o resultado desse sistema foi de 0.7 segundos, se enquadrando a situação.

6.2.3 Correlação de Pearson

A Correlação de Pearson, assim como a Similaridade Cosseno, também requer iterações através da matriz de características, resultando em uma complexidade de $O(n)$. Para um conjunto de 1000 filmes, essa métrica pode levar de 1 a 2 segundos, dependendo da implementação e da estrutura dos dados. Seu tempo de execução foi de aproximadamente 1 segundo, diferenciando do total

6.2.4 Clusterização (DBSCAN)

A clusterização utilizando DBSCAN pode ser mais intensiva em termos de recursos computacionais. A complexidade pode variar entre $O(n \log n)$ e $O(n^2)$, dependendo do valor de ϵ e do número de amostras mínimas. Para 1000 filmes, a execução pode levar de 1 a 3 segundos, variando com os parâmetros utilizados e a densidade dos dados. Essa execução foi de 2s

6.2.5 Similaridade Total

A função de Similaridade Total agrega os resultados de todas as métricas, resultando em uma complexidade total semelhante à do Índice de Jaccard, ou seja, $O(n^2)$. Para 1000 filmes, o tempo de execução pode variar de 3 a 6 segundos, dependendo das métricas utilizadas e da forma como os resultados são combinados. Sua execução também se encaixa sendo feita em 3s.

6.3 Resultados de Precisão

Para medir os resultados de precisão foi utilizado a junção dos 3 databases (que originou o `merged_df`), além da métrica de similaridade total para calculos.

A função implementada, calculou a média das avaliações dos filmes com base nas métricas de recall e precisão, permitindo assim a identificação dos resultados correspondentes a cada filme. Esse procedimento é crucial para mensurar a qualidade do sistema de recomendação, pois fornece uma análise quantitativa do desempenho das recomendações geradas. Ao combinar essas métricas, é possível obter uma visão mais abrangente sobre a eficácia do sistema e pensar em melhorias.

6.3.1 Resultados de Precisão

No nosso sistema, a precisão média foi de 0.81, indicando que a maioria das recomendações feitas pelo sistema foi pertinente. É importante notar que sistemas híbridos de recomendação geralmente alcançam precisões que variam entre 0.7 e 0.85, como observado por (Jannach e Adomavicius 2016). Assim, a precisão do nosso sistema está dentro da média, o que é satisfatório, embora não seja superior às precisões apresentadas por outros sistemas analisados. A tabela 2 apresenta uma comparação das precisões entre os sistemas analisados:

Tabela 2 – Comparação de Precisão entre Sistemas

Sistema	Precisão
Nosso Sistema	0.81
ULUYAGMUR et al. (2012)	0.213
Dwivedi & Islam (2023)	0.85
Alsekait et al. (2024)	0.98
Setiawan & Arsyntania (2024)	0.8695

Fonte: Elaborado pelo autor

6.3.2 Resultados de Recall

O recall médio do nosso sistema foi de apenas 0.07, o que indica que, apesar de sermos precisos em nossas recomendações, o sistema não está capturando uma quantidade significativa de filmes que poderiam ser relevantes para os usuários. A tabela 3 apresenta uma comparação dos recalls entre dois sistemas que forneceram essa métrica, sendo eles o de filtragem baseado em conteúdo (14) e o Alsekait(1), que utiliza filtragem híbrida. Essa tabela ilustra que o nosso sistema tem um recall de 0.07, inferior ao de ULUYAGMUR et al. (2012), que apresentou um recall de 0.095, mas significativamente mais baixo do que o recall de 0.85 do sistema de Alsekait et al. (2024). Essa discrepância destaca uma limitação crítica na capacidade do nosso sistema de capturar recomendações relevantes.

Tabela 3 – Comparação de Recall entre Sistemas

Sistema	Recall
Nosso Sistema	0.07
ULUYAGMUR et al. (2012)	0.095
Alsekait et al. (2024)	0.85

Fonte: Elaborada pelo autor

7 Conclusão

Em síntese, os resultados obtidos evidenciam que o nosso sistema de recomendação possui uma capacidade significativa de sugerir filmes relevantes, com uma média de precisão de 0.81. Essa eficácia indica que, na maioria das vezes, as recomendações geradas são pertinentes para os usuários. No entanto, o baixo recall de 0.07 ressalta uma limitação crítica: o sistema não consegue captar uma quantidade expressiva de filmes que poderiam ser do interesse dos usuários.

Essa distância entre o valor da precisão e do recall aponta para a necessidade de melhorias no sistema. Para aumentar o recall, é fundamental considerar ações como a ampliação das características dos filmes incluídas no processo, como enredos, temas e outros atributos que influenciam as preferências do público, além de oferecer um número maior de recomendações, pois com apenas 5 recomendações, o sistema acaba não mostrando filmes que podem ser de possível gosto do usuário. A aplicação de algoritmos mais avançados e a combinação de diferentes métodos de recomendação também podem contribuir para ampliar a diversidade das sugestões.

Portanto, enquanto o sistema demonstra um bom desempenho em recomendações relevantes, o aprimoramento do recall é essencial para proporcionar uma experiência mais abrangente e satisfatória aos usuários. Buscar um equilíbrio entre precisão e recall não apenas melhorará a qualidade das recomendações, mas também aumentará a satisfação dos usuários ao explorar um conjunto mais extenso de opções.

7.1 Perspectivas futuras

O sistema de recomendação desenvolvido possui grande potencial para evoluir por meio da implementação de melhorias e novas funcionalidades. A seguir, destacam-se algumas perspectivas futuras que podem aprimorar tanto a precisão quanto o recall das recomendações, além de expandir o alcance e a usabilidade do sistema.

7.1.1 Implementação de novos algoritmos e interligação de métricas

Uma das principais evoluções previstas para o sistema é a integração de novos algoritmos que explorem técnicas mais avançadas, como redes neurais ou aprendizado profundo, para identificar padrões ainda mais complexos nos dados. Além disso, a interligação de métricas mais sofisticadas e a inclusão de variáveis adicionais, como a utilização de diretores ou de tendências temporais ou preferências regionais, podem

contribuir significativamente para o aumento do recall, permitindo que o sistema ofereça recomendações mais abrangentes e alinhadas aos interesses do usuário.

7.1.2 Aplicação de outros métodos de avaliação

Atualmente, o sistema utiliza métricas clássicas como precisão e recall para medir seu desempenho. Entretanto, métodos complementares de avaliação, como a utilização do F1-score e NDCG (Normalized Discounted Cumulative Gain), podem oferecer insights mais completos sobre o desempenho geral do sistema, especialmente em cenários onde a relevância dos itens recomendados varia. Essa ampliação permitirá uma análise mais detalhada, facilitando ajustes mais direcionados nas recomendações.

7.1.3 Expansão para plataformas integradas

A implementação do sistema em uma rede mais acessível, como um site ou aplicativo, é uma meta essencial para expandir sua usabilidade e alcance. Ao integrar a aplicação em plataformas acessíveis, o sistema poderá interagir diretamente com os usuários finais, oferecendo uma experiência dinâmica e personalizada. Além disso, a coleta de feedback direto dos usuários possibilitará ajustes contínuos, garantindo maior aderência às expectativas dos mesmos.

7.1.4 Integração com APIs de atualização de conteúdo

Outra perspectiva importante é o uso de APIs capazes de detectar automaticamente novos filmes ou séries adicionados a bancos de dados, como o TMDB ou IMDb. Essas APIs permitirão a atualização dinâmica do sistema, garantindo que o catálogo de recomendações esteja sempre alinhado às novas tendências e lançamentos. Além disso, a padronização automática dos dados extraídos dessas fontes diminuirá o trabalho manual e aumentará a eficiência do processo.

7.1.5 Detalhamento de atualizações

Por fim, a introdução de uma página dentro do sistema detalhando atualizações para o usuário final, especialmente em casos de implantação em sites ou aplicativos, é crucial para aumentar a confiança e o engajamento. Por meio de notificações ou changelogs, os usuários poderão acompanhar melhorias realizadas no sistema, como o ajuste de algoritmos ou a inclusão de novos conteúdos. Essa transparência fortalece a relação com os usuários, destacando o compromisso com a evolução contínua do sistema.

Referências

- ALSEKAIT, D. M.; SHDEFAT, A. Y.; MOSTAFA, N. Next-generation movie recommenders: Leveraging hybrid deep learning for enhanced personalization. Natural Publishing, 2024. Disponível em: <https://www.naturalspublishing.com/files/published/421u7s7hh73z89.pdf>.
- DWIVEDI, P.; ISLAM, B. An item-based collaborative filtering approach for movie recommendation system. In: *Proceedings of the 10th International Conference on Computing, Communication and Automation (ICCCA)*. IEEE, 2023. Disponível em: <https://ieeexplore.ieee.org/abstract/document/10112338/>.
- GOMES, P. C. T. *Clustering: O que é Cluster Analysis e quais suas Aplicações?* 2024. Acessado em: 22 de Outubro de 2024. Disponível em: <https://analisemacro.com.br/data-science/clustering-o-que-e-cluster-analysis-e-quais-suas-aplicacoes/>.
- JACCARD, P. Étude comparative de la distribution florale dans une portion des alpes et du jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, v. 37, p. 547–579, 1901.
- JANNACH, D.; ADOMAVICIUS, G. *Recommender Systems: Past, Present, and Future*. [S.l.]: Springer, 2016.
- MONTEIRO, G.; CARL, H. *Entendendo DBSCAN*. 2020. Acessado em: 1 de Novembro de 2024. Disponível em: <https://medium.com/@gabrielmonteiro/entendendo-dbscan-por-gabriel-monteiro-e-hugo-carl-1234567890>.
- NGUYEN, T. T.; HUI, P. M.; HARPER, F. M.; TERVEEN, L.; KONSTAN, J. A. Exploring the filter bubble: The effect of using recommender systems on content diversity. In: *Proceedings of the 23rd International Conference on World Wide Web*. Seoul: ACM, 2014. p. 677–686.
- PEARSON, K. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, v. 58, p. 240–242, 1895.
- RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. 3rd. ed. [S.l.]: Pearson, 2016.
- SALLOUM, S.; RAJAMANTHRI, D. Implementation and evaluation of movie recommender systems using collaborative filtering. *Journal of Advances in Information Technology*, v. 12, n. 3, p. 189–196, 2021. Disponível em: <https://www.jait.us/uploadfile/2021/0719/20210719052408995.pdf>.
- SARWAR, B.; KARYPIS, G.; KONSTAN, J.; RIEDL, J. Item-based collaborative filtering recommendation algorithms. In: ACM. *Proceedings of the 10th International Conference on World Wide Web*. [S.l.], 2001. p. 285–295.
- SETIAWAN, E. B.; ARSYTANIA, I. H. Movie recommender system with cascade hybrid filtering using convolutional neural network. UAD, 2024. Disponível em: <https://eprints.uad.ac.id/65078/1/1-Movie%20Recommender%20System%20with%20Cascade%20Hybrid%20Filtering%20Using%20Convolutional%20Neural%20Network.pdf>.

TURING, A. M. Computing machinery and intelligence. *Mind*, v. 59, n. 236, p. 433–460, 1950.

ULUYAGMUR, M.; CATALTEPE, Z.; TAYFUR, E. Content-based movie recommendation using different feature sets. In: *Proceedings of the World Congress on Engineering and Computer Science*. [S.l.: s.n.], 2012. p. 17–24.