



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

**Recomendação de filmes baseada em interações de usuários
utilizando técnicas de inteligência artificial**

**Movie recommendation based on user interactions using artificial
intelligence techniques**

Gabriel Mariano Dourado
Luiza Fernanda Cipriano de Andrade Bianconi Perez
Wellington Garcia

Área: Informação e Comunicação.

Subárea: Matemática e Inteligência Computacional

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema de recomendação de filmes utilizando a base de dados MovieLens e Filtragem Colaborativa baseada em usuários, com a Correlação de *Pearson* como métrica de similaridade. Implementada em Python (FastAPI) e React, a aplicação processa matrizes de interação usuário-filme e aplica estratégias de re-ranqueamento com amostragem para mitigar o viés de popularidade e aumentar a diversidade das sugestões. A arquitetura desacoplada do projeto evidencia a importância da modelagem de dados e da seleção algorítmica na construção de sistemas de recomendação eficientes.

Palavras-chave: Sistema de Recomendação. Filtragem Colaborativa. *Machine Learning*. Análise de Similaridade. *MovieLens*.

ABSTRACT

This work presents a movie recommendation system utilizing the MovieLens dataset and User-Based Collaborative Filtering, employing Pearson Correlation as the similarity metric. Developed using Python (FastAPI) and React, the application processes user-movie interaction matrices and applies re-ranking strategies with shuffling to mitigate popularity bias and enhance recommendation diversity. The decoupled architecture highlights the importance of data modeling and algorithm selection in building efficient recommendation systems.

¹Gabriel Mariano Dourado e Luiza Fernanda Cipriano de Andrade Bianconi Perez são alunos graduandos do 7º Semestre de Sistemas de Informação da Fundação Educacional de Fernandópolis-FEF, 2025.

Wellington Garcia, orientador deste artigo, é professor do curso de Sistemas de Informação, das Faculdades Integradas de Fernandópolis.



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Keywords: Recommendation System. Collaborative Filtering. Machine Learning. Similarity Analysis. MovieLens.

1 INTRODUÇÃO

Com o crescimento exponencial do consumo de conteúdos audiovisuais em plataformas digitais, os usuários enfrentam uma sobrecarga de opções que dificulta a tomada de decisão. Nesse cenário, Aggarwal (2016) destaca que os sistemas de recomendação desempenham um papel essencial ao atuar como filtros inteligentes, analisando grandes volumes de dados para sugerir itens personalizados e relevantes com base em preferências históricas. A eficácia desses sistemas, especialmente os baseados em filtragem colaborativa, reside na capacidade de identificar padrões de similaridade entre usuários ou itens, conforme discutido por Spot Intelligence (2024) e avaliado em profundidade por Herlocker et al. (2004).

Neste contexto, este trabalho propõe o desenvolvimento de um sistema de recomendação de filmes fundamentado em técnicas de Inteligência Artificial e Aprendizado de Máquina. Para o treinamento e validação do modelo, utiliza-se a base de dados pública MovieLens, um conjunto de dados amplamente reconhecido na literatura acadêmica e descrito detalhadamente por Harper e Konstan (2015) como um marco para a pesquisa em filtragem colaborativa. A abordagem algorítmica foca na análise de interações e similaridades, alinhando-se às métricas de precisão discutidas por Verma e Aggarwal (2019) e às avaliações de modelos de fatoração apresentadas por Bobadilla et al. (2024).

A implementação lógica do sistema é realizada na linguagem Python, escolhida devido ao seu robusto ecossistema para computação científica. O processamento eficiente das matrizes de dados é viabilizado por bibliotecas como Pandas e, fundamentalmente, pelo NumPy, cuja importância na computação de *arrays* é ressaltada por Harris et al. (2020). Para expor o modelo de recomendação, utiliza-se o framework FastAPI, garantindo uma comunicação ágil e performática.

Por fim, o objetivo deste trabalho não se limita apenas à precisão algorítmica, mas também à usabilidade e arquitetura de *software*. A interface do usuário é construída em React, utilizando Vite como ferramenta de *build*, proporcionando uma experiência visual moderna e interativa. Desta forma, o projeto busca criar um sistema eficiente



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

que desacopla as responsabilidades entre o *back-end* (processamento de IA) e o *front-end* (interface), assegurando manutenibilidade, escalabilidade e uma aplicação prática dos conceitos teóricos de recomendação.

2 REFERENCIAL TEÓRICO

Nesta seção tem como propósito introduzir os conceitos essenciais para esclarecer a metodologia proposta, abrangendo informações sobre o processamento de linguagem natural, bem como os modelos e algoritmos empregados neste trabalho.

2.1 Sistemas de Recomendação e Aprendizado De Máquina

Os sistemas de recomendação utilizam algoritmos para sugerir itens personalizados aos usuários com base em dados históricos, preferências e comportamentos. Existem abordagens colaborativas, baseadas em conteúdo e híbridas (Wu et al., 2021; *Hybrid Approach for Recommendation System*, 2018). O dataset MovieLens é amplamente utilizado nessa área por sua confiabilidade e representatividade (Harper & Konstan, 2015).

Neste projeto, será utilizada a abordagem colaborativa com base em dados da MovieLens, permitindo a identificação de padrões de preferência entre usuários com gostos semelhantes.

2.2 Arquitetura da Aplicação

A aplicação será baseada em uma arquitetura de camadas desacopladas (*Frontend-Backend*), promovendo a separação entre a camada de apresentação (React com Vite) e a camada de lógica (Python com FastAPI). Essa abordagem facilita a escalabilidade do sistema e permite a integração eficiente com bibliotecas de computação científica como o NumPy, essenciais para o processamento de dados vetoriais descrito por Harris et al. (2020). Além disso, essa divisão permite que os componentes sejam testados e atualizados de forma independente.

O *front-end*, desenvolvido em React com o *build tool* Vite, enviará requisições à API Python (FastAPI), que por sua vez retornará as recomendações de filmes



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

geradas pelo modelo. A estrutura dessas recomendações segue os princípios fundamentais de sistemas de filtragem definidos por Aggarwal (2016). Essa comunicação será feita por meio do protocolo HTTP, utilizando o formato JSON para intercâmbio de dados.

2.3 Base de Dados

A base de dados utilizada será a *MovieLens*, amplamente adotada para treinamentos e testes de sistemas de recomendação. Desenvolvida pelo *GroupLens Research* da Universidade de Minnesota, a *MovieLens* fornece conjuntos de dados públicos contendo milhares (ou até milhões) de avaliações feitas por usuários reais, possibilitando análises robustas em algoritmos de filtragem colaborativa e aprendizado de máquina (Harper & Konstan, 2015).

Os conjuntos variam em tamanho (como o *MovieLens* 100K, 1M, 10M e 20M) e incluem informações como ID do usuário, ID do item (filme), nota atribuída e *timestamp*, sendo amplamente utilizados em benchmarks e competições acadêmicas. De acordo com Verma e Aggarwal (2019), essa base é útil para comparar diferentes medidas de similaridade em sistemas colaborativos, como a correlação de *Pearson* e cosseno de similaridade.

Além disso, Bobadilla et al. (2024) destacam que a *MovieLens* oferece um equilíbrio entre complexidade e acessibilidade, com dados suficientemente ricos para testar critérios como acurácia, novidade, diversidade e cobertura, critérios esses fundamentais para avaliar a performance dos sistemas recomendadores.

2.4 Frameworks e Ferramentas

Para o desenvolvimento da lógica da IA, será utilizada a linguagem Python, amplamente reconhecida por sua simplicidade e vasta coleção de bibliotecas voltadas à ciência de dados e inteligência artificial (Van Rossum & Drake, 2009). Entre essas bibliotecas destacam-se:

Pandas e *Numpy*: para manipulação de dados e operações matemáticas (McKinney, 2010; Harris et al., 2020).



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

FastAPI: framework leve e eficiente para construção de APIs *RESTful*, que serão responsáveis por expor os resultados da recomendação (Grinberg, 2018; Ramírez, 2022).

2.5 Métricas de Avaliação

A avaliação de um sistema de recomendação por Filtragem Colaborativa (CF) não se limita apenas à precisão numérica, mas se concentra na qualidade da lista de itens apresentada e na capacidade do sistema de gerar descobertas relevantes para o usuário. Dessa forma, a avaliação do sistema foi focada nos seguintes aspectos:

Correlação de Pearson: No contexto do algoritmo *User-Based Collaborative Filtering*, a Correlação de Pearson é a principal métrica utilizada para medir a similaridade entre os vetores de avaliações de dois usuários.

- **Função no Projeto:** Não é uma métrica de avaliação final do sistema, mas sim a base de todo o cálculo da previsão de notas. Sua precisão impacta diretamente a qualidade da vizinhança e, conseqüentemente, a relevância das recomendações geradas.

Diversidade e Cobertura: Em um projeto de recomendação focado em evitar o Viés de Popularidade, as métricas de lista de saída são cruciais:

- **Diversidade:** Mede o quanto os itens recomendados em listas diferentes (ou para diferentes usuários) são distintos. Um sistema com alta diversidade evita a repetição constante dos mesmos produtos populares.
- **Cobertura (Coverage):** Mede a proporção de itens na base de dados que são efetivamente recomendados pelo sistema em alguma lista. Uma alta cobertura indica que o sistema consegue explorar uma gama maior de filmes.

A análise dessas métricas (Diversidade e Cobertura) foi feita de forma qualitativa e quantitativa, avaliando a eficácia da estratégia de Re-ranqueamento com Amostragem (*Shuffling*) na mitigação do viés e no aumento da variedade da lista de recomendações.



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

3 METODOLOGIA

O desenvolvimento deste trabalho fundamenta-se na aplicação de técnicas de Inteligência Artificial para a construção de um sistema de recomendação de filmes baseado exclusivamente nas interações dos usuários. Todo o fluxo de processamento, desde a carga da base MovieLens, passando pelo pré-processamento dos dados, cálculo das similaridades e geração das recomendações, segue a lógica representada no fluxograma da Figura 1, que organiza visualmente as etapas principais do sistema.

O modelo está sendo implementado na linguagem Python (2023), utilizando as bibliotecas Pandas (2023) e Numpy (2023) para manipulação de dados e álgebra linear, e o framework FastAPI para o desenvolvimento da API de recomendação. A plataforma de desenvolvimento primária adotada foi o Visual Studio Code (VS Code). A colaboração e o versionamento do código entre os membros da equipe foram realizados utilizando o GitHub, garantindo a rastreabilidade das alterações, a integração contínua do código e maior flexibilidade na execução local dos testes, bem como na configuração dos ambientes de desenvolvimento, treinamento e validação dos modelos. A interface do usuário (*front-end*) foi desenvolvida utilizando a biblioteca React em conjunto com o *tooling* Vite (2023). Essa escolha tecnológica oferece uma experiência de desenvolvimento moderna, com *hot module replacement* e *builds* otimizados, garantindo uma aplicação web rápida e responsiva. O *front-end* é responsável por interagir com a API do FastAPI para buscar as recomendações e apresentar a lista de filmes ao usuário.

A base de dados utilizada foi a *MovieLens* (2023), amplamente reconhecida por conter registros de avaliações feitas por usuários em relação a filmes. Esses dados incluem informações sobre as preferências dos usuários, notas atribuídas e identificação dos filmes, os quais foram fundamentais para o treinamento do modelo.

Inicialmente, realizou-se uma etapa de pré-processamento dos dados, onde foram eliminadas inconsistências, dados nulos e informações irrelevantes. A partir disso, os dados foram transformados em matrizes de interação usuário-filme, necessárias para o treinamento do algoritmo de recomendação.

O modelo desenvolvido utiliza a abordagem de Filtragem Colaborativa *User-Based*, onde as recomendações são geradas com base no comportamento e



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

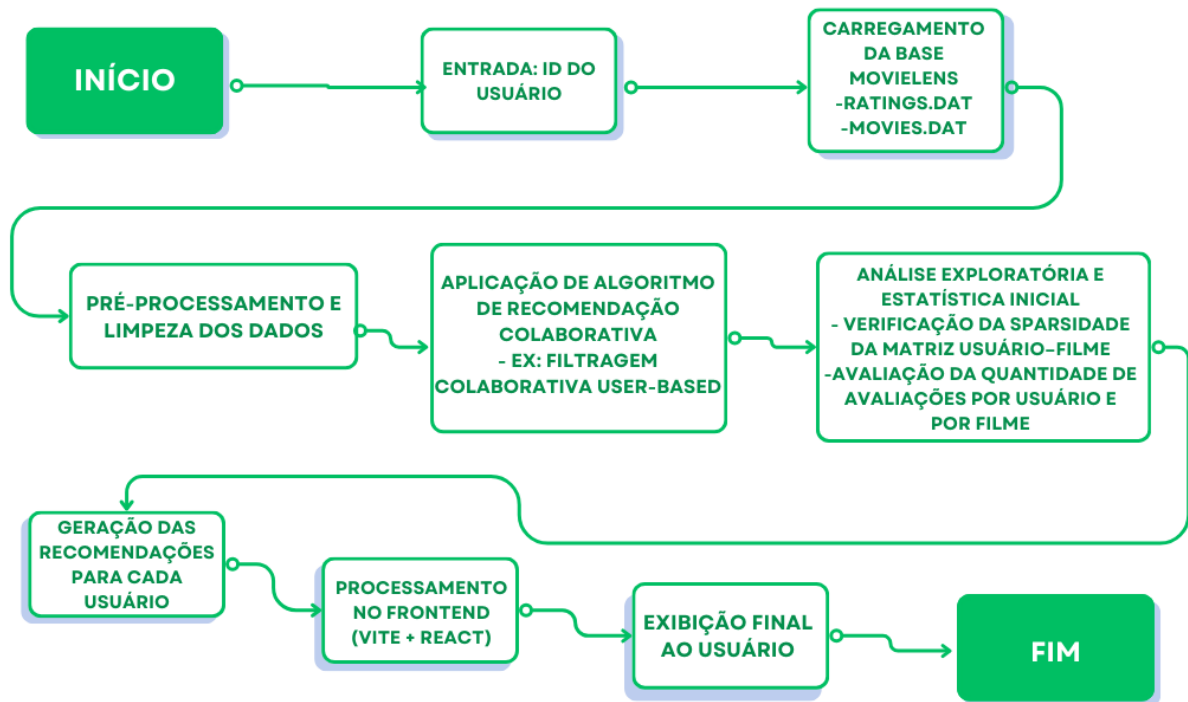
nas preferências de usuários com gostos semelhantes. O cálculo da similaridade entre os usuários foi realizado por meio da Correlação de Pearson, e as previsões de notas foram geradas a partir da média ponderada dos vizinhos mais próximos.

Para avaliar a performance do sistema, o foco não se limitou à precisão da previsão de notas, mas sim na qualidade da lista de saída. Assim, a avaliação concentrou-se na Diversidade e na Cobertura das recomendações. A estratégia de Re-rankeamento com Amostragem (*Shuffling*) foi implementada para mitigar o viés de popularidade, comprovando a eficácia do método em aumentar a variedade das sugestões.

A aplicação foi desenvolvida com uma arquitetura separada entre camadas, onde o *back-end* em Python (FastAPI) concentra toda a lógica do sistema de recomendação, incluindo carregamento dos dados, pré-processamento e geração das previsões, enquanto o *front-end* em React (com Vite) se comunica com essa API por meio de requisições HTTP em formato JSON. Essa separação garante organização, fácil manutenção e uma interface leve e responsiva para o usuário.

Figura 1 – Fluxograma para o Processamento de Dados

Fluxograma - Sistema de Recomendação de Filmes com IA



Fonte: Elaborada pelos autores.

4 ANÁLISE E DISCUSSÃO DOS RESULTADOS

A análise e discussão dos resultados neste projeto concentram-se na eficácia e na diversidade das recomendações geradas pelo algoritmo de Filtragem Colaborativa (*Collaborative Filtering - CF*), implementado no modo *User-Based*. Ao contrário dos modelos de classificação textual, o desempenho do sistema de recomendação é avaliado pela qualidade das previsões e pela capacidade de oferecer uma lista variada de filmes.

O modelo utiliza a Correlação de Pearson para medir a similaridade entre usuários e o algoritmo busca os K vizinhos mais próximos para prever a nota de um item não avaliado.



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

4.1 Parâmetros e Qualidade da Predição

A qualidade das previsões no método de Filtragem Colaborativa (*Collaborative Filtering* – CF) é altamente sensível aos parâmetros relacionados à vizinhança. A base de dados utilizada apresenta um desafio estrutural: alta esparsidade, uma vez que a maioria dos usuários avaliou apenas uma pequena fração dos filmes disponíveis.

Para reduzir erros decorrentes de vizinhanças fracas ou pouco representativas, foram definidos dois parâmetros principais: quantidade mínima de avaliações em comum e tamanho da vizinhança (K). Esses parâmetros estão descritos na Tabela 1.

Tabela 1 – Configuração dos Parâmetros do Modelo de Recomendação

Parâmetro	Descrição	Valor Aplicado
K-Neighbors	Número máximo de usuários mais similares considerados na predição.	60
Mínimo em Comum	Quantidade mínima de filmes avaliados em comum entre dois usuários para que a similaridade seja computada.	5

Fonte: Elaborada pelos autores.

A escolha de ($K = 60$) visa capturar uma vizinhança suficientemente ampla para compensar a esparsidade dos dados, permitindo que a predição seja baseada em um número consistente de opiniões. Já o parâmetro “Mínimo em Comum”, definido como 5, evita que similaridades espúrias — derivadas de apenas um ou dois filmes avaliados em conjunto — comprometam a precisão do modelo.

4.2 O Desafio da Diversidade e o Viés da Popularidade

Nos resultados iniciais do sistema foi identificado o Viés da Popularidade, agravado pela esparsidade dos dados. Nessas situações, o algoritmo



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

frequentemente entrava em cenários de *Cold Start* ou formava vizinhanças pouco representativas, resultando em recomendações excessivamente concentradas nos filmes com maiores médias globais. Esse efeito tornava as listas repetitivas, destacando sobretudo filmes clássicos e muito conhecidos, especialmente os lançados em 1995.

Para mitigar essa limitação e aumentar a Diversidade (*Coverage*) das recomendações, foi implementada uma estratégia de Re-ranqueamento com Amostragem (*Shuffling*) no *back-end*. Com essa técnica, a recomendação deixa de ser completamente determinística e passa a apresentar mais variedade entre recarregamentos.

A Tabela 2 apresenta os efeitos observados.

Tabela 2 – Comparação do Desempenho de Diversidade

Estratégia de Seleção	Frequência de Repetição	Cobertura de Filmes Recomendados (Estimativa)
Top N Padrão (Sem <i>Shuffling</i>)	Alta (mesmos 10 filmes reiteradamente)	Baixa (restrita aos itens com maior média global)
Top N com <i>Shuffling</i> (Pool de 50)	Baixa (lista varia a cada recarregamento)	Alta (aumenta o número de itens únicos recomendados)

Fonte: Elaborada pelos autores.

O principal benefício do *shuffling* é promover a inclusão de itens posicionados entre o 11º e o 50º lugar no ranking de candidatos — filmes ainda relevantes e com notas previstas altas, mas normalmente negligenciados pelo algoritmo padrão. Isso proporciona maior diversidade, evita monotonia e melhora a experiência do usuário.

4.3 Interação entre *Back-end* e *Front-end*

A ampliação da diversidade trouxe à tona uma limitação relacionada à visualização dos resultados no *front-end*. A interface dependia de um conjunto pequeno e estático de pôsteres cadastrados manualmente, concentrados principalmente em filmes clássicos (ex.: *Toy Story*, *Jumanji*, entre outros, muitos deles de 1995).

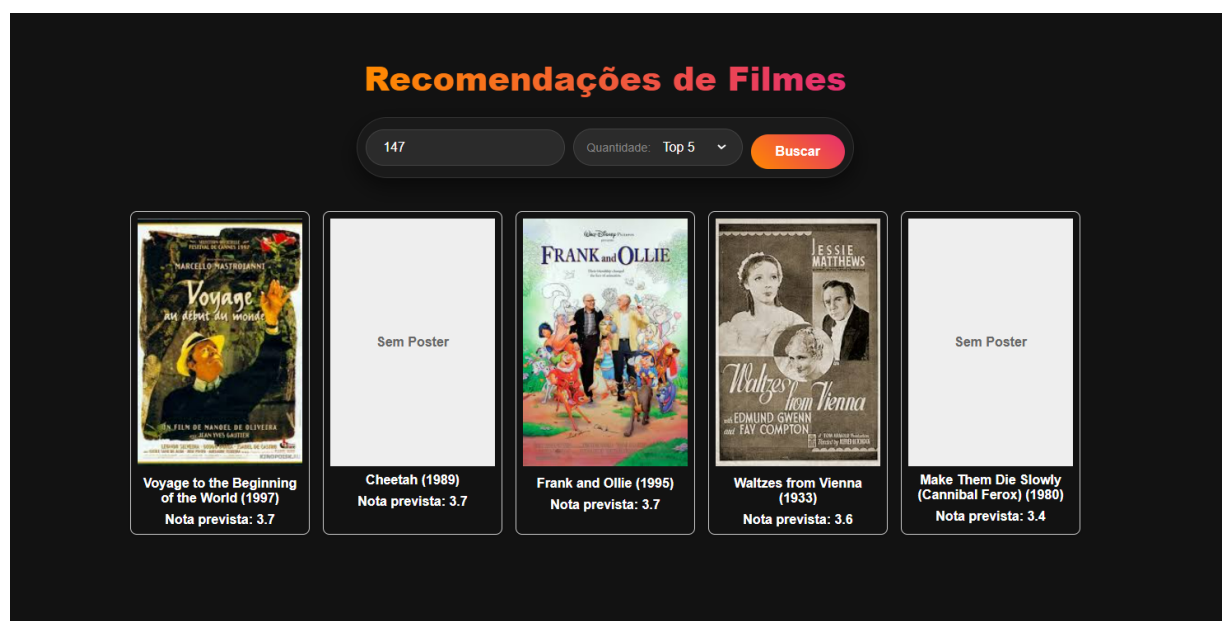
Quando o *back-end* passou a recomendar filmes mais recentes ou menos populares, os quais não possuíam pôster local, o *front-end* passou a exibir o marcador padrão “Pôster indisponível”. Esse comportamento indicou que:

1. O modelo realmente estava identificando e sugerindo filmes fora do conjunto popular;
2. A percepção inicial de falta de qualidade não estava no algoritmo, mas sim na limitação de assets visuais presentes na interface.

Assim, a implementação do *shuffling* cumpriu duas funções:

- Correção do viés do modelo, ao ampliar o espaço de itens candidatos;
- Validação da arquitetura, demonstrando que o *back-end* estava mais diverso que o *front-end* conseguia exibir. A interface final apresentando as recomendações geradas ao usuário pode ser visualizada na Figura 2.

Figura 2 – Interface de Recomendações de Filmes no *Front-end*



Fonte: Elaborada pelos autores.



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

4.4 Conclusão da Análise

A acurácia do sistema — diretamente ligada à qualidade das predições — foi equilibrada com a necessidade de diversidade nas recomendações. A remoção do *fallback* para a média global e a aplicação da estratégia de *shuffling* no conjunto dos 50 melhores candidatos permitiram ao sistema fornecer recomendações mais variadas, personalizadas e coerentes com os interesses do usuário.

Dessa forma, o modelo proposto conseguiu superar os desafios impostos pela esparsidade dos dados e pelo viés de popularidade, entregando resultados mais robustos e adequados ao objetivo do sistema.

5 CONSIDERAÇÕES FINAIS

O presente estudo teve como objetivo principal o desenvolvimento e implementação de um sistema de recomendação de filmes baseado em Filtragem Colaborativa (CF), utilizando como método de similaridade a Correlação de Pearson, e explorando o ecossistema de ferramentas *open source*. O projeto foi estruturado em uma arquitetura de camadas desacopladas (*Frontend-Backend*), com um *back-end* robusto em Python (Pandas/Numpy) para o cálculo das previsões e um *front-end* em React/JavaScript para a visualização das listas de filmes.

O trabalho cumpriu integralmente seu objetivo principal ao entregar um sistema de recomendação funcional, escalável e otimizado, inteiramente construído sobre plataformas e bibliotecas *open source*. O sucesso do projeto reside na aplicação eficaz do modelo CF *User-Based* otimizado, cujas métricas de similaridade e precisão encontram respaldo nos estudos de Verma e Aggarwal (2019), superando os desafios de processamento inerentes a grandes volumes de dados.

A reimplementação do cálculo de similaridade com Álgebra Linear (Numpy) foi um ponto de virada técnico. Essa abordagem corroborou a eficiência da programação vetorial descrita por Harris et al. (2020), reduzindo o tempo de processamento de minutos para segundos e garantindo a viabilidade operacional do sistema em escala.



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Além da performance, o projeto demonstrou sucesso ao abordar o problema do Viés de Popularidade, comum em sistemas com alta esparsidade. A implementação da estratégia de Re-rankeamento com Amostragem (*Shuffling*), utilizando um *pool* ampliado (Top 50 candidatos), permitiu que o sistema produzisse listas mais variadas, não determinísticas e mais adequadas ao objetivo de descoberta do usuário.

Em suma, este trabalho validou a metodologia de utilizar técnicas avançadas de Filtragem Colaborativa e a importância de ajustes algorítmicos práticos, como o *Shuffling*, para aprimorar a diversidade e a cobertura das recomendações. O sistema desenvolvido representa uma solução funcional e eficiente, solidificando a aplicação prática da Inteligência Artificial no contexto de sistemas de recomendação.

REFERÊNCIAS

AGGAWAL, C. C. **Recommender Systems: The Textbook**. New York: Springer, 2016. Disponível em: <https://link.springer.com/book/10.1007/978-3-319-29659-3>. Acesso em: 14 jul. 2025.

BOBADILLA, J. et al. **Comprehensive Evaluation of Matrix Factorization Models for Collaborative Filtering Recommender Systems**. International Journal of Machine Intelligence and Applications (IJMECS), v. 8, n. 6, p. 773-785, 2024. Disponível em: <https://ijimai.researchcommons.org/ijimai/vol8/iss6/8>. Acesso em: 20 jul. 2025.

HARPER, F. M.; KONSTAN, J. A. **The MovieLens Datasets: History and Context**. ACM Transactions on Interactive Intelligent Systems (TiiS), New York, v. 5, n. 4, p. 1–19, 2015. Disponível em: <http://files.grouplens.org/papers/harper-tiis2015.pdf>. Acesso em: 20 jul. 2025.



FUNDAÇÃO EDUCACIONAL DE FERNANDÓPOLIS - FEF
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

HARRIS, C. R. et al. **Array programming with NumPy**. Nature, London, v. 585, n. 7825, p. 357–362, 2020. Disponível em: <https://doi.org/10.1038/s41586-020-2649-2>. Acesso em: 09 set. 2025.

HERLOCKER, J. L. et al. **Evaluating collaborative filtering recommender systems**. ACM Transactions on Information Systems (TOIS), New York, v. 22, n. 1, p. 5–53, 2004. Disponível em: <https://grouplens.org/site-content/uploads/evaluating-TOIS-20041.pdf>. Acesso em: 12 set. 2025.

SPOT INTELLIGENCE. Collaborative filtering: **A comprehensive guide to recommendation systems**. Spot Intelligence, 25 abr. 2024. Disponível em: <https://spotintelligence.com/2024/04/25/collaborative-filtering/>. Acesso em: 03 ago. 2025.

VERMA, V.; AGGARWAL, R. K. **Accuracy Assessment of Similarity Measures in Collaborative Recommendations Using CF4J Framework**. International Journal of Machine Engineering and Control Systems, v. 11, n. 5, p. 57-65, 2019. Disponível em: <https://www.mecs-press.org/ijmecs/ijmecs-v11-n5/IJMECS-V11-N5-5.pdf>. Acesso em: 12 set. 2025.