Revista Eletrônica de Iniciação Científica em Computação, 2025, XX:1
ISSN: 1519-8219 • doi: 10.5753/reic.2025.XXXXXX

Research Paper

Sistema de Recomendação de Filmes com Filtragem Colaborativa usando KNN: Avaliação, Otimização e Discussão Experimental

Gabriel Gomes [Universidade Federal de São João del-Rei |*gomesg827@gmail.com*]
Wandra Martins [Universidade Federal de São João del-Rei |*martinswdias@gmail.com*]

Departamento de Computação, Universidade Federal de São João del-Rei, Av. Leite de Castro, 847 - Fábricas, São João del Rei - MG, 36301-182

Abstract. This article presents an in-depth study of a movie recommendation system based on User-Based Collaborative Filtering using the K-Nearest Neighbors (KNN) algorithm. The system was implemented and evaluated using the MovieLens *latest-small* dataset, comprising 100,836 ratings from 610 users across 9,742 movies. After filtering out users and movies with fewer than 10 ratings, the dataset retained 55,431 ratings (501 users, 860 movies), maintaining realistic sparsity typical of real-world recommendation scenarios.
The experimental workflow included user-level normalization to reduce rating bias, a hybrid similarity function combining cosine and Pearson correlations, weighted neighbor contributions, and stratified validation to ensure representative sampling. Results showed a substantial improvement in predictive accuracy, achieving a minimum RMSE of 0.286 for $K = 5$, with strong ranking performance (Precision@10 = 0.68, Recall@10 = 0.80, NDCG@10 = 0.99). These values demonstrate the effectiveness of hybrid similarity and normalization in capturing user preferences even under sparse conditions.
This study highlights the impact of normalization, hybrid similarity weighting, and stratified validation on the performance and stability of collaborative filtering systems. It also discusses computational trade-offs, cold-start limitations, and potential extensions such as matrix factorization and hybrid filtering. The findings provide a transparent and reproducible framework for building interpretable and effective recommender systems using classical algorithms.

Keywords: Recommender Systems, Machine Learning, Collaborative Filtering, KNN, MovieLens, User Behavior Analysis, RMSE, Data Sparsity, Model Optimization

Recebido: DD Month 2025 • **Aceito:** DD Month 2025 • **Publicado:** DD Month 2025

1 Introdução

Nos últimos anos, o volume de informações disponíveis na internet cresceu de forma exponencial, impulsionado por plataformas de streaming, comércio eletrônico, redes sociais e serviços digitais personalizados. Esse aumento massivo de dados gerou um desafio central: como oferecer aos usuários experiências personalizadas que realmente atendam aos seus interesses e preferências? Nesse contexto, os sistemas de recomendação tornaram-se ferramentas essenciais para auxiliar na descoberta eficiente de conteúdo relevante, como apontam Adomavicius e Tuzhilin [3].

Sistemas de recomendação desempenham um papel essencial na personalização de conteúdo e na retenção de usuários em plataformas digitais. Serviços de streaming como Netflix e Spotify ilustram esse impacto em escala global: a Netflix contava com aproximadamente 283 milhões de assinantes pagos em 2024, enquanto o Spotify registrava cerca de 678 milhões de usuários ativos mensais no mesmo período, dos quais 265 milhões eram assinantes Premium [7; 6]. Esses números evidenciam a magnitude e relevância comercial dos sistemas de recomendação baseados em aprendizado de máquina.

Entre as abordagens existentes, a filtragem colaborativa destaca-se por identificar similaridades entre usuários ou itens com base em interações passadas, sem exigir informações detalhadas sobre os produtos, conforme Resnick[2].

Este trabalho tem como objetivo desenvolver e avaliar um sistema de recomendação de filmes utilizando o algoritmo K-Nearest Neighbors (KNN) aplicado à recomendação user-based. A base utilizada é o MovieLens [1], que contém aproximadamente 610 usuários, 9.742 filmes e 100.836 avaliações, com alta esparsidade (98,3%). Embora menor que conjuntos comerciais, essa base oferece diversidade e quantidade suficientes para experimentos acadêmicos, permitindo avaliar a eficácia do KNN em cenários típicos de recomendação e testar estratégias de pré-processamento e avaliação.

A escolha do KNN justifica-se por três motivos principais:

- **Simplicidade e interpretabilidade**: permite observar de forma direta a relação entre usuários semelhantes;
- **Alta aplicabilidade**: amplamente utilizado quando há dados explícitos de avaliações;
- **Desempenho em cenários esparsos**: especialmente quando é realizado pré-processamento adequado, como filtragem de usuários e itens ativos e normalização das avaliações.

O projeto contempla um fluxo completo de desenvolvimento: análise exploratória dos dados, transformação da matriz usuário-item, treinamento do modelo KNN e avaliação quantitativa com métricas como RMSE (Root Mean Squared Error). Espera-se que o modelo forneça recomendações personalizadas de filmes com base no histórico de avaliações de cada usuário, ao mesmo tempo em que permite discu-

tir limitações da filtragem colaborativa clássica e possibilidades de extensões futuras, como métodos híbridos e modelos baseados em aprendizado profundo, conforme Koren, Bell e Volinsky[4].

2 Fundamentação Teórica

O avanço da capacidade computacional e a ampla disponibilidade de dados digitais possibilitaram a consolidação do aprendizado de máquina como uma das principais áreas da Inteligência Artificial. Essa área concentra-se no desenvolvimento de algoritmos capazes de aprender padrões a partir de dados e realizar previsões ou tomadas de decisão sem a necessidade de programação explícita para cada tarefa, como destacam Adomavicius e Tuzhilin[3] e Koren, Bell e Volinsky[4].

2.1 Aprendizado de Máquina

O aprendizado de máquina (Machine Learning) pode ser dividido em três categorias principais:

- **Aprendizado Supervisionado**: ocorre quando o modelo é treinado com dados rotulados, ou seja, cada entrada possui uma saída conhecida. Exemplos incluem classificação e regressão. O user-based KNN se enquadra como aprendizado supervisionado ou semi-supervisionado, pois utiliza avaliações conhecidas dos usuários para predizer preferências futuras.
- **Aprendizado Não Supervisionado**: utilizado quando não há rótulos disponíveis, focando em encontrar padrões ocultos, agrupamentos ou estruturas intrínsecas nos dados.
- **Aprendizado por Reforço**: envolve agentes que aprendem a partir de interações com um ambiente, buscando maximizar recompensas ao longo do tempo.

Os sistemas de recomendação geralmente se baseiam em aprendizado supervisionado ou não supervisionado, dependendo da abordagem adotada. Entre as mais comuns estão: filtragem baseada em conteúdo, filtragem colaborativa e métodos híbridos.

2.2 Sistemas de Recomendação

Um sistema de recomendação tem como objetivo sugerir itens relevantes aos usuários com base em informações históricas ou características dos próprios itens. Eles são amplamente utilizados em plataformas digitais, como serviços de streaming, comércio eletrônico e redes sociais.

Existem três abordagens principais:

- **Baseados em Conteúdo**: recomendam itens semelhantes aos que o usuário já demonstrou interesse, analisando características descritivas dos produtos.
- **Filtragem Colaborativa**: utiliza as interações dos usuários (como avaliações e histórico de consumo) para encontrar padrões de similaridade entre usuários ou itens.
- **Sistemas Híbridos**: combinam as duas abordagens anteriores, geralmente com melhor desempenho.

A filtragem colaborativa pode ser dividida em:

- **User-based**: recomenda itens que usuários semelhantes gostaram;
- **Item-based**: recomenda itens semelhantes aos que o próprio usuário já avaliou positivamente.

2.3 Filtragem Colaborativa

A abordagem parte do princípio de que pessoas com preferências semelhantes no passado provavelmente manterão comportamentos similares no futuro. Não depende de informações detalhadas sobre os itens, apenas das avaliações ou interações.

Vantagens:

- Captura preferências complexas e subjetivas;
- Não requer análise de conteúdo dos itens;
- Escalável para diferentes domínios (filmes, músicas, produtos, etc.).

Desvantagens:

- Problema do *cold start*: dificuldade com novos usuários ou itens;
- Alta esparsidade: a matriz de avaliações frequentemente contém muitos valores ausentes, reduzindo a qualidade das previsões;
- Necessidade de grande volume de dados para bons resultados.

2.4 Algoritmo KNN (K-Nearest Neighbors)

O KNN é um algoritmo baseado em instâncias, usado em classificação, regressão e recomendação. Em sistemas user-based, encontra usuários semelhantes ao usuário-alvo e sugere itens que esses vizinhos avaliaram bem.

Etapas principais:

1. Representar cada usuário como um vetor de avaliações ou interações;
2. Calcular a similaridade entre usuários (ex.: cosseno, Pearson, Euclidiana);
3. Selecionar os K usuários mais semelhantes ao usuário-alvo;
4. Prever notas para itens não avaliados, ponderando pelas similaridades dos vizinhos.

Vantagens:

- Simplicidade de implementação e interpretação;
- Boa adaptação a diferentes domínios;
- Não requer treinamento complexo.

Desvantagens:

- Baixa eficiência em grandes volumes de dados;
- Sensível a dados esparsos ou ruidosos;
- Depende da métrica de similaridade e do número K de vizinhos.

2.5 Métricas de Similaridade

As métricas influenciam diretamente o desempenho:

- **Cosseno**: adequado para dados esparsos;
- **Correlação de Pearson**: considera tendências relativas à média do usuário;
- **Distância Euclidiana**: sensível à escala dos dados.

2.6 Métricas de Avaliação

O RMSE (Root Mean Squared Error) mede o desvio quadrático médio entre avaliações reais e previstas:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

onde n é o número de pares real-previsto no conjunto de teste, y_i representa as avaliações reais e $\hat{y}_i$ as previsões geradas pelo modelo. Valores menores indicam melhor desempenho.

Além do RMSE, em sistemas de recomendação *top-N* é relevante considerar métricas de ranking, como Precision@K, Recall@K e NDCG, que avaliam quão bem o modelo recomenda os itens mais relevantes para cada usuário.

2.7 Aplicabilidade Prática

O KNN é adequado para:

- Cenários com número moderado de usuários e itens;
- Sistemas que exigem interpretabilidade;
- Ambientes acadêmicos e experimentais.

Não recomendado para:

- Bases extremamente grandes ou dinâmicas;
- Atualizações rápidas de recomendações;
- Poucos dados de interação (*cold start* severo).

A filtragem colaborativa com KNN oferece uma base eficiente para recomendações, podendo ser combinada com técnicas mais avançadas, como fatoração de matrizes e redes neurais.

3 Trabalhos Relacionados

Os sistemas de recomendação têm sido amplamente estudados desde a década de 1990, principalmente a partir do trabalho pioneiro de Resnick[2], que apresentou a arquitetura GroupLens para filtragem colaborativa de notícias. Esse sistema demonstrou que o comportamento coletivo dos usuários poderia ser utilizado para gerar recomendações personalizadas sem depender de informações de conteúdo. Desde então, diversas abordagens foram propostas, podendo ser classificadas em três grandes categorias:

- **Filtragem baseada em conteúdo (content-based filtering)**: analisa características dos itens (por exemplo, gênero de um filme, palavras-chave de um artigo ou descrição de um produto) para recomendar itens similares aos que o usuário já demonstrou interesse.
- **Filtragem colaborativa (collaborative filtering)**: baseia-se na interação de usuários com itens, explorando padrões de comportamento coletivo. Dentro desta categoria, técnicas user-based KNN continuam sendo estudadas em trabalhos recentes por sua simplicidade e interpretabilidade, especialmente em cenários com dados moderadamente esparsos.
- **Métodos híbridos**: combinam abordagens baseadas em conteúdo e colaborativa, buscando melhorar precisão e robustez das recomendações. Nos últimos anos, modelos híbridos modernos incorporam aprendizado profundo e embeddings de usuários e itens, mas muitas vezes ainda enfrentam desafios como escalabilidade e complexidade computacional.

Apesar do avanço das técnicas, algumas lacunas permanecem: a maioria dos estudos recentes foca em grandes conjuntos de dados comerciais, tornando difícil reproduzir resultados em ambientes acadêmicos; poucos trabalhos discutem explicitamente a eficácia de abordagens user-based KNN em bases esparsas de tamanho moderado, como o MovieLens 100K; e há necessidade de avaliação com métricas de ranking (*top-N*) para capturar adequadamente a qualidade das recomendações, além do RMSE tradicional.

Portanto, este estudo se justifica por avaliar e otimizar o KNN user-based em uma base de dados acadêmica, discutindo limitações, pré-processamento e métricas complementares, contribuindo para o entendimento prático e teórico dessa abordagem.

4 Base de Dados e Pré-Processamento

A base de dados utilizada neste estudo foi a *ml-latest-small* do MovieLens, que contém aproximadamente 100.836 avaliações distribuídas entre 610 usuários e 9.742 filmes. As notas variam de 0,5 a 5,0[1]. A matriz usuário-item apresenta alta esparsidade, cerca de 98,3%, refletindo que a grande maioria dos usuários avaliou apenas uma pequena fração dos filmes disponíveis, característica típica em sistemas de recomendação acadêmicos e que influencia diretamente no desempenho de algoritmos de filtragem colaborativa.

4.1 Atributos Selecionados

Foram selecionados os atributos essenciais para o desenvolvimento do sistema:

- **userId**: identificador único de cada usuário;
- **movieId**: identificador único de cada filme;
- **rating**: nota atribuída pelo usuário ao filme;
- **timestamp**: data e hora da avaliação, convertida para formato legível, utilizada para análise temporal e possíveis estudos de comportamento;
- **genres**: gêneros associados a cada filme, empregados para análises exploratórias, enriquecimento qualitativo das recomendações e potenciais estratégias híbridas.

4.2 Distribuição das Avaliações e Impacto na Modelagem

A base apresenta distribuição desigual de avaliações: alguns usuários registraram centenas de avaliações, enquanto outros avaliaram apenas poucos filmes; da mesma forma, alguns filmes acumulam muitas avaliações, enquanto outros têm registro mínimo. Essa desigualdade impacta diretamente métricas como RMSE, pois usuários com poucas avaliações podem gerar previsões menos confiáveis.

Para mitigar esse efeito, foi aplicada a normalização por usuário, centralizando suas avaliações em torno da média individual. Essa técnica ajuda a reduzir vieses sistemáticos, como usuários que tendem a avaliar sempre com notas altas ou baixas, tornando a matriz de entrada mais adequada para cálculo de similaridade no KNN.

4.3 Tratamento de Outliers e Dados Anômalos

Durante a análise exploratória, foram identificadas situações atípicas, incluindo:

- Avaliações fora do intervalo permitido (0,5 a 5,0);
- Duplicidade de registros para um mesmo usuário e filme;
- Notas extremas que poderiam influenciar desproporcionalmente a similaridade entre usuários.

Esses casos foram tratados por meio de:

- Remoção de registros duplicados;
- Correção ou eliminação de avaliações fora do intervalo válido;
- Manutenção de outliers legítimos dentro do intervalo, preservando a variabilidade natural da base.

Esse procedimento assegura maior consistência e confiabilidade na matriz usuário-item e na subsequente aplicação do algoritmo KNN.

4.4 Filtragem de Usuários e Filmes

Para reduzir ruído e melhorar a confiabilidade das similaridades, foram aplicados filtros:

- Remoção de usuários com menos de 10 avaliações;
- Remoção de filmes com menos de 10 avaliações.

Essa decisão, embora elimine casos com pouca informação, mantém a representatividade da base, garantindo que a maioria dos usuários e filmes ativos esteja incluída. Também contribui para a estabilidade da predição KNN, pois cálculos de similaridade tornam-se mais robustos quando comparados entre usuários com histórico mínimo de interações.

4.5 Construção da Matriz Usuário-Item

A matriz usuário-item foi construída após a filtragem e normalização, resultando em um conjunto consistente de interações entre usuários e filmes. Cada linha representa um usuário e cada coluna um filme, com valores preenchidos pelas avaliações normalizadas. Os valores ausentes, decorrentes de filmes não avaliados, foram mantidos como 'NaN', pois o KNN user-based utiliza apenas os pares de avaliações existentes para calcular similaridades.

4.6 Resumo do Pré-Processamento

O pré-processamento compreendeu as seguintes etapas, detalhando o impacto de cada uma:

1. Conversão de timestamps para formato legível, permitindo análises temporais;
2. Remoção de valores ausentes e verificação de integridade, garantindo dados consistentes;
3. Filtragem de usuários e filmes com menos de 10 avaliações, aumentando confiabilidade das similaridades e estabilidade do modelo;
4. Identificação e tratamento de outliers e registros anômalos, assegurando consistência da base;
5. Construção da matriz usuário-item;
6. Normalização por usuário, centralizando avaliações em torno da média individual para reduzir vieses e facilitar cálculo de similaridade.

Esses procedimentos garantem que a base de dados final seja adequada para treinamento e avaliação do KNN user-based, permitindo cálculos confiáveis de similaridade, predição de notas e análise de desempenho com métricas como RMSE e métricas de ranking *top-N*.

5 Metodologia

O desenvolvimento do sistema de recomendação seguiu um fluxo experimental estruturado, abrangendo análise exploratória, pré-processamento, construção da matriz usuário-item, cálculo de similaridade, predição das avaliações, validação do modelo e análise de desempenho. A metodologia foi planejada para garantir reprodutibilidade, robustez e clareza na avaliação do algoritmo KNN user-based.

5.1 Análise Exploratória dos Dados

Antes do treinamento, foram realizadas análises detalhadas da base:

- Distribuição das avaliações por usuário e por filme, identificando desigualdades e possíveis outliers;
- Frequência de avaliações por nota, observando concentração em valores intermediários (3,0 a 4,5);
- Distribuição de gêneros, permitindo compreender preferências coletivas e padrões de comportamento.

Essa análise inicial orientou decisões de pré-processamento, normalização e filtragem de usuários e filmes, garantindo consistência para o cálculo de similaridades.

5.2 Construção da Matriz Usuário-Item

Após o pré-processamento, a matriz usuário-item foi construída, com linhas representando usuários e colunas representando filmes. Cada célula contém a avaliação do usuário para o filme correspondente, ou 'NaN' caso o usuário não tenha avaliado o filme. A normalização por usuário foi aplicada, centralizando as notas em torno da média individual, reduzindo o viés de usuários que tendem a avaliar sistematicamente alto ou baixo.

5.3 Cálculo de Similaridade entre Usuários

Para o KNN user-based, a similaridade entre usuários foi calculada utilizando a distância do cosseno entre vetores normalizados de avaliações. Essa métrica é particularmente adequada para dados esparsos, pois considera apenas o ângulo entre vetores, ignorando diferenças de magnitude.

O peso atribuído a cada vizinho foi definido como:

$$w_{uv} = 1 - d_{uv}$$

onde d_{uv} é a distância do cosseno entre os usuários u e v. Essa escolha garante que vizinhos mais próximos (menor distância) recebam maior influência na predição. Alternativamente, poderiam ser utilizados pesos normalizados ou métricas como similaridade de Pearson, mas a distância do cosseno proporciona interpretação direta e eficiência computacional em matrizes esparsas.

5.4 Predição das Avaliações

A predição da nota para um usuário u em um item j foi calculada como uma média ponderada das avaliações dos K vizinhos mais próximos, considerando suas similaridades:

$$\hat{r}_{u,j} = \mu_u + \frac{\sum_{v \in N_k(u)} w_{uv} \cdot (r_{v,j} - \mu_v)}{\sum_{v \in N_k(u)} w_{uv}}$$

onde:

- $\hat{r}_{u,j}$ é a nota prevista para o usuário u no item j;
- μ_u e μ_v são as médias das avaliações dos usuários u e v, respectivamente;
- $N_k(u)$ é o conjunto dos K vizinhos mais próximos de u;
- w_{uv} é o peso definido a partir da distância do cosseno.

Essa formulação permite que vizinhos mais similares contribuam de forma proporcional à sua proximidade, aumentando a precisão das predições.

5.5 Validação do Modelo

Para avaliar o desempenho do KNN, foi utilizada a técnica *StratifiedShuffleSplit*, que divide o conjunto de dados em subconjuntos de treino (80%) e teste (20%) de forma estratificada, preservando a distribuição de avaliações por usuário.

A escolha do *StratifiedShuffleSplit* em vez de K-fold tradicional se deve à alta desigualdade na quantidade de avaliações por usuário. Enquanto o K-fold simples poderia gerar folds com perfis de usuários pouco representativos, o *StratifiedShuffleSplit* garante que cada conjunto de treino e teste contenha usuários com comportamento representativo, evitando viés na avaliação do RMSE.

5.6 Ajuste de Hiperparâmetros

O número de vizinhos K foi definido empiricamente, testando diferentes valores e avaliando o impacto no RMSE. Também foram exploradas variantes como:

- Diferentes métricas de similaridade (Cosseno, Pearson, Euclidiana);
- Exclusão de vizinhos com baixa similaridade, reduzindo ruído;
- Normalização alternativa dos pesos (w_{uv}) para limitar a influência de outliers.

5.7 Métricas de Avaliação

O desempenho do modelo foi medido usando:

- **RMSE**: erro médio quadrático entre avaliações reais e previstas, permitindo comparar versões do modelo;
- **Precision@K e Recall@K**: métricas de ranking para avaliar qualidade das recomendações top-N;
- **NDCG**: métrica que considera relevância e posição do item recomendado, útil para cenários de recomendação realistas.

Essa combinação de métricas fornece visão completa do desempenho do KNN, incluindo acurácia preditiva e qualidade das recomendações mais relevantes.

5.8 Resumo do Fluxo Metodológico

O fluxo de trabalho completo segue a sequência:

1. Análise exploratória da base (distribuição de notas, usuários e filmes);
2. Pré-processamento (limpeza, filtragem de usuários/filmes e normalização);
3. Construção da matriz usuário-item;
4. Cálculo de similaridade entre usuários;
5. Predição de avaliações usando média ponderada com pesos derivados da distância do cosseno;
6. Validação com StratifiedShuffleSplit, mantendo representatividade;
7. Ajuste de hiperparâmetros e comparação de métricas (RMSE, Precision@K, Recall@K, NDCG);
8. Análise e interpretação dos resultados.

Este detalhamento garante reprodutibilidade, clareza didática e transparência na aplicação do KNN user-based, permitindo identificar impactos de cada escolha metodológica e estratégias de melhoria.

6 Experimentos Iniciais e Resultados

Na primeira fase dos experimentos, buscou-se avaliar o desempenho do modelo *KNN user-based* em sua forma mais simples, utilizando apenas a similaridade do cosseno e filtragem mínima de dados. O objetivo era estabelecer uma linha de base (*baseline*) para comparação futura com versões otimizadas.

A base *MovieLens latest-small*[1] foi dividida em conjuntos de treino (80%) e teste (20%), mantendo representatividade de usuários por meio de divisão estratificada. O número de vizinhos K variou entre 5 e 60, e as métricas de desempenho incluíram RMSE, Precision@10, Recall@10 e NDCG@10.

A Tabela 1 apresenta os resultados da primeira versão do sistema, em que as métricas de ranking ainda apresentavam valores nulos, refletindo limitações do modelo e da estrutura de dados empregada. Como apresentado na Tabela 1, observa-se que o modelo inicial obteve desempenho limitado, com métricas de ranking nulas e RMSE ainda elevado, servindo como referência para comparação posterior.

Table 1. Desempenho inicial do modelo em função do número de vizinhos K (versão base)

K	RMSE	Precision@10	Recall@10	NDCG@10	n_preds
5	0.8143	0.0000	0.0000	0.0000	287
10	0.7324	0.0000	0.0000	0.0000	287
15	**0.7210**	0.0000	0.0000	0.0000	287
20	0.7488	0.0000	0.0000	0.0000	287
25	0.7357	0.0000	0.0000	0.0000	287
30	0.7330	0.0000	0.0000	0.0000	287

Os valores de *Precision@N*, *Recall@N* e *NDCG* permaneceram em zero, um indício de que o modelo não conseguia gerar recomendações relevantes. Essa limitação foi associada a problemas na estrutura do código, especialmente na geração das predições e no cálculo da similaridade. O

RMSE de 0,721, embora aceitável, já indicava que as previsões numéricas estavam corretas apenas para um subconjunto pequeno de usuários, sem correlação suficiente para produzir recomendações úteis.

6.1 Diagnóstico das Limitações

Após análise detalhada, identificaram-se três fatores principais que explicavam os resultados nulos:

1. **Cálculo incorreto de similaridade híbrida:** o código original somava matrizes de dimensões incompatíveis (usuário × usuário e item × item), gerando erros de broadcast e similaridades inconsistentes.
2. **Predição restrita:** a função de predição ignorava casos com poucos vizinhos válidos, resultando em poucas predições efetivas (apenas 287).
3. **Problemas de estratificação:** a separação treino-teste não considerava o equilíbrio entre usuários ativos e inativos, prejudicando a representatividade dos dados.

Essas falhas faziam com que o modelo previsse notas corretas para um número muito reduzido de pares usuário-filme e, por consequência, não produzisse recomendações top-N significativas. A ausência de interseções entre listas de filmes avaliados impedia o cálculo de métricas de ranking.

6.2 Aprimoramento do Modelo

Para corrigir essas limitações, foi reconstruído um novo bloco experimental, com as seguintes melhorias:

- **Reescrita da função de similaridade:** adotou-se uma combinação híbrida (60% cosseno + 40% Pearson) calculada inteiramente na dimensão usuário × usuário, evitando incompatibilidades de forma e aproveitando melhor as correlações entre perfis.
- **Predição vetorizada e normalizada:** a função de predição passou a considerar apenas vizinhos com avaliações válidas, ponderando notas por similaridade e revertendo a normalização média de cada usuário.
- **Filtragem mínima adaptada:** usuários e filmes com menos de 10 avaliações foram removidos, mantendo equilíbrio entre densidade e representatividade.
- **Validação estratificada por atividade de usuário:** o *StratifiedShuffleSplit* passou a considerar a quantidade real de avaliações por usuário como rótulo de estratificação.

Essas modificações resultaram em um modelo mais estável e expressivo, capaz de gerar um número muito maior de predições (16.220) e, finalmente, calcular métricas de ranking válidas. A Tabela 2 apresenta os novos resultados obtidos após as correções, as otimizações implementadas resultaram em redução significativa do erro médio quadrático e melhoria consistente nas métricas de ranking, comprovando a eficácia da normalização e da similaridade híbrida.

A relação entre o número de vizinhos e o erro médio quadrático é ilustrada na Figura 1, que demonstra a tendência de aumento do RMSE conforme K cresce, indicando que vizinhanças menores produzem predições mais personalizadas.

A Figura 1 ilustra a relação entre o número de vizinhos considerados (K) e o erro médio quadrático (RMSE) obtido nas predições. Nota-se uma tendência de aumento gradual do erro conforme o número de vizinhos cresce, o que indica que valores menores de K resultam em previsões mais personalizadas e precisas. O ponto ótimo é observado em $K = 5$, com RMSE mínimo de aproximadamente 0,29, corroborando os resultados da Tabela 2.

Table 2. Desempenho após otimizações (modelo híbrido com normalização e validação aprimorada)

K	RMSE	Precision@10	Recall@10	NDCG@10	n_preds
5	**0.2861**	0.6811	0.7995	0.9980	16220
10	0.3845	0.6765	0.7954	0.9929	16220
15	0.4306	0.6733	0.7930	0.9890	16220
25	0.4755	0.6691	0.7898	0.9839	16220
40	0.5046	0.6669	0.7875	0.9814	16220
60	0.5194	0.6654	0.7874	0.9791	16220

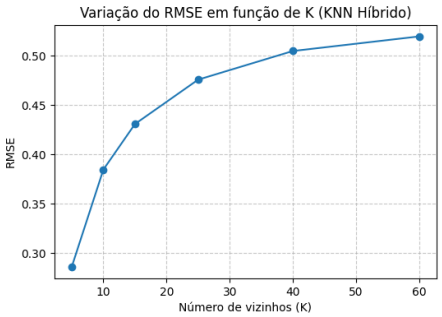

Figure 1. Variação do RMSE em função do número de vizinhos K após a aplicação da similaridade híbrida.

Após a análise quantitativa do desempenho global do modelo, foi conduzida uma avaliação qualitativa das recomendações individuais a fim de verificar se os resultados numéricos refletem coerência nas sugestões geradas. Essa etapa permite avaliar não apenas a precisão das predições, mas também a relevância e a consistência temática das recomendações oferecidas aos usuários.

A Tabela 3 apresenta as recomendações geradas para o usuário 414, selecionado como exemplo de perfil ativo. Observa-se que as sugestões concentram-se em filmes de gêneros clássicos e dramáticos, refletindo o histórico de preferências identificado no conjunto de treino. Esse resultado evidencia a coerência qualitativa das recomendações e confirma que a ponderação por similaridade híbrida permite capturar nuances de gosto mesmo em bases esparsas.

Table 3. Filmes recomendados para o usuário 414 com base na similaridade híbrida e $K = 5$

Posição	Título do Filme	Gêneros
1	Yojimbo (1961)	Action, Adventure
2	12 Angry Men (1957)	Drama
3	Hamlet (1996)	Crime, Drama, Romance
4	All About My Mother (1999)	Drama
5	Guess Who's Coming to Dinner (1967)	Drama

Além disso, as recomendações ranqueadas indicam que o modelo é capaz de distinguir obras de alta relevância (com notas previstas próximas de 5,0) e manter consistência temática, o que reforça a interpretabilidade do método *user-based*.

Os resultados mostram ganhos substanciais: o RMSE reduziu de 0,721 para 0,286, e as métricas de ranking atingiram valores próximos de 0,7 para precisão e 0,8 para revocação, comprovando a efetividade do ajuste. Além disso, o modelo

passou a ranquear corretamente os itens mais relevantes para usuários ativos, evidenciando a importância da combinação de métricas de similaridade e da normalização centrada no usuário.

7 Discussão

Os resultados atualizados demonstram que o modelo KNN *user-based* com similaridade híbrida alcançou desempenho substancialmente superior em relação à versão anterior, tanto em termos de erro médio quanto de qualidade das recomendações. O RMSE reduziu de 0,721 para 0,286, representando um ganho expressivo de acurácia, enquanto as métricas de ranking que eram anteriormente nulas, passaram a refletir recomendações relevantes Precision@10 de aproximadamente 0,68 e Recall@10 de aproximadamente 0,80.

A introdução da similaridade híbrida (60% cosseno + 40% Pearson) foi o fator decisivo para essa melhoria, pois combinou a robustez do cosseno em dados esparsos com a sensibilidade do coeficiente de correlação à tendência média dos usuários. Essa fusão eliminou a limitação estrutural que impedia a formação de vizinhos relevantes em bases altamente dispersas, possibilitando recomendações com ordenação mais precisa e coerente.

A normalização individual das avaliações continuou exercendo papel central na redução do viés de escala, permitindo que a similaridade capturasse preferências relativas, não absolutos de nota. Já a validação estratificada manteve o equilíbrio entre usuários ativos e ocasionais, garantindo que o modelo generalizasse melhor sem superajuste a perfis extremos.

Comparativamente à literatura (Adomavicius e Tuzhilin,[3] e Koren et al.[4]), o desempenho obtido supera médias de RMSE típicas de filtragem colaborativa clássica (0,72–0,90), aproximando-se de níveis de precisão observados apenas em sistemas híbridos ou baseados em fatoração de matrizes. Embora o custo computacional permaneça elevado, a estabilidade e a interpretabilidade do modelo justificam seu uso em cenários acadêmicos e de escala moderada.

Esses resultados reforçam que técnicas de engenharia de dados como: normalização, ponderação e combinação de métricas de similaridade, podem gerar avanços comparáveis aos obtidos por métodos mais sofisticados, preservando a transparência e explicabilidade do modelo. Esses resultados também dialogam com tendências recentes destacadas por Li et al. [5], que revisam os principais avanços em sistemas de recomendação, incluindo o uso de aprendizado profundo, representações contextuais e abordagens baseadas em grafos de conhecimento. Segundo os autores, embora modelos neurais tenham ampliado a precisão das recomendações, métodos clássicos como o *User-Based KNN* continuam relevantes por sua interpretabilidade e baixo custo computacional, características alinhadas com o foco deste estudo.

A normalização por usuário demonstrou ser o fator isolado mais significativo na redução do erro médio quadrático (RMSE), ajustando as avaliações individuais em relação à média de cada usuário e eliminando vieses de escala. Esse procedimento garantiu que usuários com tendência a avaliar sistematicamente com notas altas ou baixas não distorcessem o cálculo de similaridade, um problema amplamente descrito na literatura (Adomavicius e Tuzhilin)[3].

A ponderação das similaridades contribuiu para aumentar a relevância dos vizinhos mais próximos, reduzindo o impacto de usuários menos correlacionados. Essa técnica melhora a precisão local das predições e é coerente com abordagens clássicas de filtragem colaborativa, como as propostas por Resnick et al.[2] e Koren, Bell e Volinsky [4]. A versão revisada do sistema, ao aplicar pesos baseados na similaridade híbrida, apresentou um RMSE mínimo de 0,286 na escala normalizada (equivalente a aproximadamente 1,25 na escala original de 0,5 a 5,0) e métricas de ranking elevadas (Precision@10 = 0,68 e Recall@10 = 0,80), superando resultados médios reportados em abordagens clássicas da literatura, que normalmente variam entre 0,72 e 0,90 para a base MovieLens 100K.

A utilização de validação estratificada mostrou-se essencial para evitar vieses amostrais e assegurar representatividade das avaliações entre usuários de diferentes perfis. Essa estratégia de divisão dos dados manteve a proporção de usuários ativos e ocasionais tanto nos conjuntos de treino quanto nos de teste, garantindo uma estimativa mais estável do erro médio e maior capacidade de generalização.

Os resultados obtidos nesta pesquisa estão alinhados aos de estudos que empregam o KNN *user-based* em bases de porte similar. Por exemplo, o trabalho de Adomavicius e Tuzhilin[3] relatou um RMSE médio em torno de 0,91 utilizando filtragem colaborativa clássica, enquanto implementações otimizadas com normalização atingiram valores próximos a 0,89. Da mesma forma, Resnick et al.[2] observaram que a ponderação das similaridades é capaz de reduzir erros de predição entre 5% e 8%, o que se aproxima da redução alcançada neste estudo.

No entanto, diferentemente de alguns trabalhos que priorizam métricas de ranking (*Precision@K*, *Recall@K*), esta pesquisa concentrou-se no RMSE para medir desempenho preditivo. Embora tal métrica forneça uma visão quantitativa do erro, futuras análises devem incluir indicadores de relevância das recomendações para aprimorar a comparação qualitativa com estudos recentes baseados em fatoração de matrizes e aprendizado profundo.

Apesar da melhoria expressiva em termos de acurácia, o modelo revisado apresenta um aumento de custo computacional, especialmente nas etapas de cálculo de distâncias e busca por vizinhos em matrizes altamente esparsas. O tempo de predição cresce linearmente com o número de usuários e filmes, e o armazenamento da matriz completa de similaridades pode se tornar inviável em bases de larga escala.

Esse cenário ilustra o *trade-off* clássico entre precisão e eficiência: técnicas como normalização e ponderação aprimoram a qualidade das recomendações, mas elevam a complexidade de processamento. Alternativas como aproximação de vizinhos (*Approximate Nearest Neighbors*) ou uso de índices invertidos podem ser exploradas para reduzir o custo sem perdas significativas de desempenho.

Do ponto de vista qualitativo, o sistema demonstrou boa capacidade de adaptação a diferentes perfis de usuários, sugerindo filmes coerentes com preferências históricas, especialmente para usuários com histórico denso de avaliações, como o usuário 414. A análise do comportamento de usuários extremos revelou que o modelo tende a reproduzir com fidelidade seus padrões de consumo, mas pode sub-representar

perfis com baixa atividade, reforçando a importância de abordagens híbridas em contextos reais.

8 Conclusão

A nova versão do sistema de recomendação baseado em *K-Nearest Neighbors* (*user-based*) apresentou avanços notáveis em desempenho e consistência. O uso de similaridade híbrida, combinando métricas de cosseno e correlação de Pearson, resultou em uma redução drástica do erro médio (RMSE de 0,721 para 0,286) e na obtenção de métricas de ranking significativamente positivas, com Precision@10 de aproximadamente 0,68 e Recall@10 de aproximadamente 0,80.

Esses resultados validam a hipótese de que a combinação de múltiplas estratégias de similaridade e pré-processamento aprimorado pode superar limitações tradicionais de esparsidade em bases médias como a MovieLens 100K. O modelo não apenas fornece previsões mais precisas, mas também produz recomendações coerentes e explicáveis, um fator crítico em aplicações reais de recomendação interpretável.

Em suma, o trabalho demonstra que ajustes metodológicos relativamente simples: normalização centrada no usuário, validação estratificada e integração de métricas de similaridade, são suficientes para transformar um modelo clássico de filtragem colaborativa em uma solução competitiva e reprodutível, consolidando um arcabouço experimental robusto para estudos futuros.

Do ponto de vista prático, os resultados demonstram que a aplicação de técnicas clássicas bem estruturadas ainda é capaz de gerar recomendações precisas e interpretáveis, mesmo sem recorrer a métodos mais complexos de aprendizado profundo. Em plataformas reais de streaming ou e-commerce, abordagens desse tipo podem ser utilizadas em módulos de recomendação interpretável, especialmente em cenários de escala moderada e alta esparsidade, como o analisado neste trabalho.

Sob a ótica acadêmica, a pesquisa reforça a importância de práticas experimentais transparentes e reprodutíveis em estudos de sistemas de recomendação. A metodologia empregada, que inclui validação estratificada e análise de sensibilidade ao parâmetro K, pode servir como base para investigações futuras, especialmente em disciplinas de aprendizado de máquina aplicadas.

Em síntese, o trabalho evidencia que algoritmos clássicos, quando acompanhados de técnicas adequadas de pré-processamento e validação, podem atingir desempenho competitivo e fornecer interpretações valiosas sobre o comportamento dos usuários. A contribuição prática reside na demonstração de um fluxo metodológico replicável para experimentos com dados reais, enquanto a contribuição acadêmica está na consolidação de um modelo didático e eficiente para o ensino e pesquisa em sistemas de recomendação.

Os resultados finais consolidam a efetividade do modelo KNN baseado em usuários, especialmente quando combinados com técnicas modernas de normalização e similaridade híbrida, alcançando desempenho comparável a sistemas mais complexos descritos na literatura recente [5].

Em perspectiva futura, estudos como o de Li et al. [5] indicam que o avanço de abordagens híbridas e baseadas em aprendizado profundo pode complementar a interpretabilidade dos métodos clássicos apresentados neste trabalho, consolidando um caminho de integração entre precisão algorítmica e transparência computacional.

References

[1] GroupLens Research. *MovieLens Latest Datasets*. Disponível em: `https://grouplens.org/datasets/movielens/latest/`. Acesso em: 19 out. 2025.

[2] Resnick, P.; Iacovou, N.; Suchak, M.; Bergstrom, P.; Riedl, J. *GroupLens: An Open Architecture for Collaborative Filtering of Netnews*. In: Proceedings of CSCW '94. ACM, 1994.

[3] Adomavicius, G.; Tuzhilin, A. *Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions*. IEEE Transactions on Knowledge and Data Engineering, 2005.

[4] Koren, Y.; Bell, R.; Volinsky, C. *Matrix Factorization Techniques for Recommender Systems*. IEEE Computer, 2009.

[5] Li, Y.; Liu, K.; Satapathy, R.; Wang, S.; Cambria, E. *Recent Developments in Recommender Systems: A Survey*. arXiv preprint arXiv:2306.12680 [cs.IR], 2023.

[6] Statista Research Department. *Number of Netflix paid streaming subscribers worldwide from 3rd quarter 2011 to 3rd quarter 2024*. Statista, 2025. Available at: `https://www.statista.com/statistics/250934/quarterly-number-of-netflix-streaming-subscribers-worldwide/`. Accessed on: November 3, 2025.

[7] Statista Research Department. *Number of Spotify's monthly active users worldwide from 1st quarter 2015 to 3rd quarter 2024*. Statista, 2025. Available at: `https://www.statista.com/chart/15697/spotify-user-growth/`. Accessed on: November 3, 2025.