Universidade Federal da Paraíba
Centro de Ciências Sociais Aplicadas
Graduação em Ciência de Dados para Negócios

RecomendaAí: Do Pipeline de Dados à Recomendação Personalizada em Aplicação Web

Lorena Ximenes Carlos

João Pessoa - PB
2026

Lorena Ximenes Carlos

RecomendaAí: Do Pipeline de Dados à Recomendação Personalizada em Aplicação Web

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Ciência de Dados para Negócios do Centro de Ciências Sociais Aplicadas da Universidade Federal da Paraíba (UFPB), como requisito para obtenção do grau de Bacharel em Ciência de Dados para Negócios.

Orientador: Jorge Henrique Norões Viana

João Pessoa - PB
2026

Catalogação na publicação
Seção de Catalogação e Classificação

C284r Carlos, Lorena Ximenes.
RecomendaAí: do pipeline de dados à recomendação personalizada em aplicação web / Lorena Ximenes Carlos. - João Pessoa, 2026.
52 f. : il.

Orientação: Jorge Henrique Norões Viana.
TCC (Graduação) - UFPB/CCSA.

1. Sistemas de recomendação. 2. Filtragem baseada em conteúdo. 3. Similaridade de cosseno. 4. TF-IDF. 5. Sistemas web. I. Viana, Jorge Henrique Norões. II. Título.

UFPB/CCSA CDU 004.6:330(043)

Elaborado por ANDRE DOMINGOS DA SILVA FERNANDES - CRB-15/00730

Lorena Ximenes Carlos

RecomendaAí: Do Pipeline de Dados à Recomendação Personalizada em Aplicação Web

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Ciência de Dados para Negócios do Centro de Ciências Sociais Aplicadas da Universidade Federal da Paraíba (UFPB), como requisito para obtenção do grau de Bacharel em Ciência de Dados para Negócios.

Trabalho aprovado. João Pessoa - PB, 11 de agosto de 2026:

Jorge Henrique Norões Viana
Orientador

Prof. Dr. Ignácio Tavares de Araújo Junior
Examinador

Prof. Dr. José Jorge Lima Dias Jr.
Examinador

João Pessoa - PB
2026

Agradecimentos

Agradeço, em primeiro lugar, a Deus, sem o qual nada disso teria sido possível.

À minha família, pelo apoio incondicional em todas as minhas decisões, inclusive na escolha deste curso, e por estarem presentes nos momentos bons e nos momentos difíceis.

Às amigas que a faculdade me deu: sem elas, esse caminho teria sido muito mais difícil.

Ao meu orientador, pela paciência, pelo incentivo e pelo acompanhamento cuidadoso na construção deste trabalho.

Por fim, agradeço aos membros da banca examinadora por aceitarem avaliar este projeto.

Resumo

O RecomendaAí é uma aplicação web de recomendação de filmes e séries desenvolvida no contexto de Ciência de Dados aplicada, com foco na mitigação da sobrecarga de escolha em catálogos audiovisuais extensos. A solução utiliza dados reais provenientes da API do The Movie Database (TMDb), consolidados em um catálogo processado de 18.128 títulos, para gerar recomendações em um ambiente integrado com exploração de catálogo, autenticação, personalização e registro de interações dos usuários. A metodologia combina filtragem baseada em conteúdo, por meio de vetorização TF-IDF (*Term Frequency–Inverse Document Frequency*) e similaridade de cosseno, com busca semântica opcional por *embeddings* e estratégias de reordenação híbrida fundamentadas em sinais comportamentais e no contexto de uso. A aplicação organiza a experiência nas abas Buscar, Descobrir e Para Você, permitindo recomendação a partir de títulos conhecidos, consultas em linguagem natural e personalização contínua a partir do histórico de interações. O trabalho demonstra a integração de técnicas de Ciência de Dados e de desenvolvimento web na construção de um sistema de recomendação funcional, interpretável e utilizável, capaz de reduzir o espaço de busca do usuário diante de catálogos extensos.

Palavras-chave: Sistemas de recomendação. Filtragem baseada em conteúdo. Similaridade de cosseno. TF-IDF. Sistemas web.

Abstract

The RecomendaAí is a web application for movie and TV series recommendation developed in the context of Applied Data Science, focusing on mitigating choice overload in extensive audiovisual catalogs. The solution uses real data from The Movie Database (TMDb) API, consolidated into a processed catalog of 18,128 titles, to generate recommendations within an integrated environment that includes catalog exploration, authentication, personalization, and user interaction tracking. The methodology combines content-based filtering through TF-IDF (*Term Frequency–Inverse Document Frequency*) vectorization and cosine similarity, with optional semantic search via embeddings and hybrid reranking strategies based on behavioral signals and usage context. The application organizes the user experience into the Search, Discover, and For You tabs, enabling recommendations from known titles, natural-language queries, and continuous personalization from interaction history. This work demonstrates the integration of Data Science techniques and web development in building a functional, interpretable, and usable recommendation system capable of reducing the user's search space in large catalogs.

Keywords: Recommender systems. Content-based filtering. Cosine similarity. TF-IDF. Web systems.

Lista de tabelas

Lista de ilustrações

Sumário

1 Introdução

O consumo audiovisual doméstico deixou de ser organizado por uma grade de programação e passou a ser organizado por catálogos. No Brasil, os serviços de vídeo por assinatura pela internet alcançaram 33,4 milhões de domicílios em 2025, o equivalente a 44,4% dos lares com televisão, segundo o módulo de Tecnologias da Informação e Comunicação da Pesquisa Nacional por Amostra de Domicílios Contínua (Instituto Brasileiro de Geografia e Estatística, 2026). Além de crescer, a oferta se fragmentou: levantamento da Comscore sobre televisão conectada indica que o domicílio brasileiro declara acessar, em média, 8,9 serviços de *streaming* por mês, sendo 4,6 pagos e 4,4 gratuitos, com consumo diário concentrado entre 19h e meia-noite e forte predominância de filmes e séries entre os formatos assistidos (Comscore, 2026). O espectador contemporâneo, portanto, não escolhe dentro de um catálogo, mas entre vários catálogos simultâneos, cada um com sua própria interface, seus próprios critérios de organização e seu próprio mecanismo de sugestão.

A dimensão desses catálogos é o segundo elemento do problema. Análise da Gracenote, divulgada no relatório *State of Play* da Nielsen, contabilizou 1,9 milhão de títulos de vídeo disponíveis aos espectadores de cinco mercados (Estados Unidos, Reino Unido, Canadá, México e Alemanha) em julho de 2021, número que chegou a 2,7 milhões em junho de 2023, dos quais aproximadamente 86,7% acessíveis por serviços de *streaming* (Nielsen, 2023). No mercado brasileiro, o Panorama do Mercado de Vídeo por Demanda da Ancine registra dezenas de plataformas em operação, com catálogos individuais na ordem de milhares de obras: a Amazon Prime Video, por exemplo, com mais de 7,6 mil títulos, e a Netflix com cerca de 5,2 mil (Agência Nacional do Cinema, 2023). Trata-se de uma quantidade de alternativas que nenhum usuário é capaz de percorrer, comparar ou sequer inspecionar de forma exaustiva antes de decidir o que assistir.

O efeito prático dessa abundância é o aumento do custo cognitivo da escolha. Já em 2004, Schwartz argumentava que o excesso de alternativas, em vez de ampliar a liberdade do consumidor, tende a elevar a ansiedade e reduzir a satisfação com a decisão tomada, fenômeno conhecido como paradoxo da escolha (Schwartz, 2004). Os dados de audiência sugerem que o diagnóstico se materializou no *streaming*: o tempo médio gasto por sessão apenas decidindo o que assistir passou de pouco mais de sete minutos, em 2019, para cerca de 10,5 minutos em 2023, e uma parcela relevante dos espectadores relata desistir da sessão e fazer outra coisa quando não encontra algo que considere adequado (Nielsen, 2023). Evidências qualitativas apontam ainda que a sobrecarga não decorre apenas do tamanho absoluto do catálogo, mas também da forma como as opções são apresentadas: listas de recomendação densas, repetitivas ou pouco diversificadas podem intensificar, e não aliviar, o esforço de decisão (Meza; D'Urso, 2024). Em outras palavras, o problema é

simultaneamente de volume de dados e de mediação: alguém, ou algum sistema, precisa reduzir o espaço de busca antes que o usuário o encontre.

É nesse contexto que se insere o RecomendaAí, uma aplicação web de recomendação de filmes e séries desenvolvida como fluxo completo de Ciência de Dados. A solução parte de um catálogo real obtido pela API do TMDb, passa por pré-processamento e análise exploratória, representa os itens por TF-IDF e por *embeddings* semânticos, calcula similaridade de cosseno e aplica um *ranking* híbrido sensível ao contexto e às interações do usuário. O resultado é disponibilizado em interface web com abas de busca, exploração e personalização, priorizando interpretabilidade e reprodutibilidade, de modo que seja possível compreender como metadados textuais e sinais comportamentais influenciam as recomendações geradas.

Este trabalho está organizado da seguinte forma. O Capítulo de Objetivos apresenta o objetivo geral e os objetivos específicos. Em seguida, a Revisão da Literatura fundamenta os conceitos de sobrecarga informacional, filtragem baseada em conteúdo, representação vetorial, sistemas híbridos, *cold start* e busca semântica. O Capítulo de Dados caracteriza a fonte TMDb, a amostra e a análise exploratória do catálogo processado, estabelecendo as condições empíricas que orientam as escolhas técnicas. A Metodologia descreve o pré-processamento, a vetorização, a personalização e o *ranking* híbrido. A Arquitetura do Sistema detalha as camadas da solução. Os Resultados apresentam a aplicação em operação, com exemplos práticos das abas Buscar, Descobrir e Para Você. Por fim, as Considerações Finais sintetizam contribuições, limitações e possibilidades de trabalhos futuros.

2 Objetivos

2.1 Objetivo Geral

Desenvolver e disponibilizar uma aplicação web de recomendação de filmes e séries, baseada em filtragem por conteúdo e sinais comportamentais, capaz de gerar sugestões personalizadas e relevantes a partir de dados dos conteúdos reais e interações dos usuários.

2.2 Objetivos Específicos

- Coletar, consolidar e caracterizar dados de filmes e séries obtidos pela API do TMDb, identificando limitações de cobertura, idioma e completude textual relevantes para a modelagem.
- Preparar representações textuais dos itens a partir de gêneros, palavras-chave e sinopses, com tratamento de valores ausentes, ruído linguístico e atributos inconsistentes.
- Implementar e analisar um recomendador baseado em conteúdo com TF-IDF e similaridade de cosseno para busca por título, mantendo-o também como mecanismo de contingência da busca textual.
- Implementar uma busca semântica por *embeddings* para consultas em linguagem natural e comparar qualitativamente seus resultados com o *fallback* TF-IDF.
- Desenvolver um *ranking* híbrido que combine, em escala comum, similaridade de conteúdo, afinidade de gênero, popularidade, recência e recorrência entre títulos âncora, com pesos sensíveis ao contexto.
- Integrar o recomendador a uma aplicação web com funcionalidades de busca, exploração, personalização e persistência de interações, utilizando *onboarding* e itens populares para mitigar o *cold start*.
- Avaliar o comportamento do protótipo por meio de consultas e perfis ilustrativos, examinando coerência temática, limitações e efeito dos pesos contextuais sobre a ordenação.

3 Revisão da Literatura

Esta seção apresenta o referencial teórico que fundamenta o desenvolvimento do RecomendaAí, organizado em torno de cinco eixos temáticos: sobrecarga informacional e sistemas de recomendação; filtragem baseada em conteúdo; representação vetorial e recuperação de informação; filtragem colaborativa e sistemas híbridos; e estratégias para *cold start* e busca semântica.

3.1 Sobrecarga informacional e sistemas de recomendação

O fenômeno da sobrecarga de escolha em ambientes com grande volume de opções foi sistematizado por Schwartz (2004), que argumenta que o excesso de alternativas, ao contrário de ampliar a liberdade do consumidor, tende a aumentar a ansiedade, reduzir a satisfação e dificultar a tomada de decisão. Embora essa discussão tenha sido realizada em um contexto geral do comportamento do consumidor, seus efeitos tornam-se ainda mais evidentes no cenário atual, em que plataformas de *streaming* disponibilizam catálogos com milhares de filmes e séries, ampliando os desafios relacionados à descoberta de conteúdos relevantes.

Estudos recentes detalham como essa sobrecarga se manifesta concretamente no consumo audiovisual digital. Meza e D'Urso (2024), em estudo qualitativo com usuários da Netflix, mostram que a abundância de títulos, combinada a listas de recomendação densas e pouco diversificadas, tende a prolongar o tempo de busca, elevar o esforço cognitivo da escolha e gerar reações emocionais negativas, como frustração e insatisfação moderada com o conteúdo selecionado. Os autores descrevem ainda um "dilema do usuário": embora os espectadores confiem e dependam fortemente das listas recomendadas para navegar o catálogo, essa dependência não elimina a sobrecarga; em alguns casos, pode até intensificá-la quando as sugestões são percebidas como pouco atrativas ou repetitivas. Assim, no contexto de plataformas *over-the-top* (OTT), a sobrecarga não decorre apenas do tamanho absoluto do catálogo, mas também da forma como as opções são apresentadas e filtradas na interface.

Como resposta a esse desafio, surgiram os sistemas de recomendação. Resnick e Varian (1997), em um dos trabalhos seminais da área, definiram esses sistemas como ferramentas capazes de auxiliar usuários na identificação de informações ou itens relevantes em grandes coleções de dados, com base em critérios explícitos ou implícitos. Desde então, essas técnicas evoluíram significativamente e passaram a desempenhar papel central em plataformas digitais, oferecendo recomendações personalizadas em diferentes domínios, embora, como evidenciam Meza e D'Urso (2024), sua eficácia em reduzir a sobrecarga

dependa do equilíbrio entre personalização, diversidade e qualidade percebida das sugestões.

Ricci, Rokach e Shapira (2015) ampliam essa definição ao tratar sistemas de recomendação como uma infraestrutura de filtragem de informação capaz de aprender preferências individuais e projetá-las sobre catálogos extensos, reduzindo o espaço de busca e aumentando a probabilidade de satisfação do usuário. Revisões mais recentes reiteram essa centralidade e consolidam taxonomias atualizadas de modelos, técnicas e domínios de aplicação (Ko *et al.*, 2022). Do ponto de vista da indústria, o impacto econômico desses sistemas motivou décadas de pesquisa sobre como modelar preferências, representar itens e combinar diferentes sinais para produzir sugestões relevantes (Adomavicius; Tuzhilin, 2005).

Entre as dimensões discutidas por Adomavicius e Tuzhilin (2005), destaca-se a recomendação sensível a contexto, em que fatores como tempo, localização ou estado do usuário influenciam a relevância dos itens sugeridos. Trabalhos posteriores mantêm essa linha e exploram, por exemplo, a incorporação de viés contextual em modelos de fatoração matricial (Casillo *et al.*, 2022). Essa perspectiva é retomada na metodologia do RecomendaAí, tanto na forma de pesos de *ranking* ajustados por horário e por preferência explícita de momento quanto em seleções editoriais apresentadas na página inicial da aplicação.

Esses trabalhos, contudo, não permitem concluir que toda forma de contextualização reduz automaticamente a sobrecarga. A recomendação pode estreitar o espaço de busca, mas listas excessivamente homogêneas reforçam a superespecialização, enquanto listas muito diversas transferem novamente ao usuário o esforço de escolha. O problema de projeto não é, portanto, maximizar apenas similaridade ou variedade, mas administrar esse compromisso. No RecomendaAí, essa leitura fundamenta listas curtas, diversidade guiada e exposição do contexto aplicado; como não há experimento controlado com usuários, a redução efetiva do esforço cognitivo permanece uma hipótese sustentada pelo desenho do produto, e não uma relação causal demonstrada neste estudo.

3.2 Filtragem baseada em conteúdo e representação vetorial

A filtragem baseada em conteúdo é uma das abordagens mais clássicas e interpretáveis de recomendação. Nessa estratégia, itens são representados por suas características descritivas, como gênero, sinopse e palavras-chave, e usuários ou consultas são comparados com essas representações para recuperar os itens mais semelhantes (Pazzani; Billsus, 2007). Revisões contemporâneas organizam o campo em taxonomias atualizadas de técnicas baseadas em conteúdo, incluindo o uso de TF-IDF e medidas de similaridade vetorial (Papadakis *et al.*, 2023).

Lops, Gemmis e Semeraro (2011) descrevem o ciclo típico: representação do item

por atributos extraídos de sua descrição, construção de um perfil do usuário com base em interações passadas e comparação do perfil com o catálogo para gerar recomendações ranqueadas. Os autores destacam como principal vantagem a interpretabilidade, uma vez que é possível explicar os motivos pelos quais um item foi recomendado com base em atributos específicos, bem como a independência em relação aos dados de outros usuários. Entre as limitações, destacam a tendência à superespecialização, em que o sistema recomenda apenas itens muito similares ao histórico, reduzindo a diversidade.

A base matemática para essa abordagem é o modelo espaço-vetorial introduzido por Salton e McGill (1983), no qual documentos e consultas são representados como vetores em um espaço de termos, permitindo calcular similaridade por operações algébricas. O ponderador mais utilizado nesse modelo é o TF-IDF (*Term Frequency–Inverse Document Frequency*), cuja componente de frequência inversa de documentos foi formalizada por Jones (1972). O IDF reduz o peso de termos que aparecem em muitos documentos e, portanto, são pouco discriminativos, enquanto amplifica termos raros e específicos. Em aplicações recentes de recomendação cinematográfica, o TF-IDF permanece amplamente empregado para representar sinopses e metadados textuais antes do cálculo de similaridade (Permana; Wibowo, 2023).

Manning, Raghavan e Schütze (2008) demonstram que a similaridade de cosseno é a medida mais adequada para vetores de texto por ser invariante ao comprimento dos documentos, propriedade especialmente relevante em catálogos com sinopses de tamanhos muito variados. Essa escolha permanece consolidada em aplicações contemporâneas: em sistemas recentes de recomendação de filmes baseada em conteúdo, a similaridade de cosseno continua sendo empregada para comparar vetores TF-IDF derivados de sinopses e metadados textuais (Permana; Wibowo, 2023), e também é a métrica padrão para comparar *embeddings* semânticos de sentenças no Sentence-BERT (Reimers; Gurevych, 2019) e em modelos contrastivos posteriores, como o SimCSE (Gao; Yao; Chen, 2021).

O TF-IDF e os *embeddings* respondem a necessidades distintas. O primeiro preserva correspondências lexicais, é esparso, auditável e pouco custoso, mas não aproxima adequadamente sinônimos ou paráfrases. Os *embeddings* ampliam a recuperação conceitual e multilíngue, porém tornam a justificativa de cada resultado menos transparente e exigem modelo e artefatos adicionais. Assim, não se assume superioridade universal de uma representação: a adequação depende da tarefa. Essa distinção sustenta o TF-IDF na recomendação a partir de um título conhecido e os *embeddings* como modo principal da consulta livre, com TF-IDF como contingência operacional.

3.3 Filtragem colaborativa, sistemas híbridos e *cold start*

A filtragem colaborativa utiliza o comportamento coletivo de usuários para produzir recomendações, com a premissa de que usuários com históricos similares tendem a apreciar

os mesmos itens no futuro. Koren, Bell e Volinsky (2009) descrevem as técnicas de fatoração matricial que se tornaram o padrão da área após a competição Netflix Prize[1], demonstrando alta capacidade de capturar fatores latentes de preferência. Revisões posteriores confirmam a continuidade dessas técnicas e discutem seus algoritmos, aplicações e desafios específicos (Isinkaye, 2021). Sua principal limitação é a necessidade de grande volume de interações, o que a torna inadequada para sistemas com base de usuários reduzida.

Burke (2002) identificou que as limitações individuais de cada abordagem podem ser mitigadas pela combinação em sistemas híbridos, propondo uma taxonomia de estratégias que inclui ponderação de *scores*, troca entre modelos e mistura de resultados. Surveys recentes reforçam que abordagens híbridas seguem sendo uma das principais estratégias para equilibrar precisão e cobertura em diferentes domínios (Ko *et al.*, 2022). O RecomendaAí adota uma forma próxima da ponderação, combinando similaridade de conteúdo com sinais comportamentais em uma função de *ranking* única.

A ponderação oferece transparência e controle, mas exige que os sinais estejam em escalas comparáveis; caso contrário, a variável de maior amplitude pode dominar o resultado independentemente do peso declarado. Essa condição é especialmente importante ao combinar similaridade, contagem de votos e recência. Por isso, o *ranking* implementado normaliza cada componente no conjunto de candidatos antes da soma ponderada. A opção por essa solução determinística também reflete a ausência de logs em volume suficiente para treinar e validar um modelo colaborativo ou de aprendizagem de pesos, limitação coerente com o estágio de protótipo e com o problema de *cold start*.

Relacionado a essa limitação, destaca-se o problema de *cold start*, que ocorre quando há informações insuficientes sobre novos usuários ou novos itens, dificultando a identificação de padrões de preferência e a geração de recomendações relevantes. Como os sistemas de recomendação dependem de dados históricos para inferir interesses, a ausência dessas informações compromete a qualidade das sugestões iniciais. Schein *et al.* (2002) apontam como estratégias para mitigar esse problema a coleta de preferências explícitas durante o processo de *onboarding* e a utilização de itens populares como mecanismo de contingência. Revisões sistemáticas mais recentes consolidam essas e outras estratégias, orientadas a dados ou a algoritmos, propostas na última década para mitigar o *cold start* (Panda; Ray, 2022).

Por fim, para superar a limitação do TF-IDF em capturar relações semânticas entre palavras, Reimers e Gurevych (2019) apresentaram o *Sentence-BERT*[2], modelo que gera *embeddings* de sentenças capazes de preservar proximidade semântica independentemente do vocabulário superficial, sendo utilizado no RecomendaAí como representação principal

[1] Competição promovida pela Netflix entre 2006 e 2009 com o objetivo de incentivar o desenvolvimento de algoritmos de recomendação capazes de melhorar em pelo menos 10% a precisão do sistema utilizado pela plataforma na época.

[2] Extensão do modelo BERT (*Bidirectional Encoder Representations from Transformers*) que produz *embeddings* de sentenças semanticamente significativos e comparáveis por similaridade de cosseno.

da busca em linguagem natural quando o artefato pré-computado está disponível. Desenvolvimentos posteriores, como o SimCSE, mantêm a similaridade de cosseno como métrica central para comparar representações densas de sentenças (Gao; Yao; Chen, 2021). Neste trabalho, TF-IDF e *embeddings* não têm seus escores somados: os modos são alternativos, porque representam relações diferentes e produzem distribuições de similaridade não diretamente comparáveis. Uma fusão de listas, por exemplo por *Reciprocal Rank Fusion*, permanece como possibilidade de avaliação futura mediante conjunto de consultas com julgamentos de relevância.

3.4 Síntese da revisão

A Tabela 1 apresenta uma síntese dos principais conceitos discutidos na revisão da literatura e sua relação com o desenvolvimento do RecomendaAí. Os conceitos foram organizados em três dimensões: o tema abordado, a principal referência utilizada como fundamentação teórica e sua respectiva aplicação no contexto do sistema proposto. Essa organização permite visualizar de forma consolidada como os elementos teóricos estudados contribuíram para as decisões de projeto e implementação adotadas neste trabalho.

Tabela 1 – Síntese dos conceitos e aplicações no RecomendaAí

Conceito	**Referência principal**	**Aplicação no RecomendaAí**
Sobrecarga de escolha	Schwartz (2004); Romero Meza e D'Urso (2024)	Motivação do problema no contexto de *streaming*
Sistemas de recomendação	Resnick e Varian (1997); Ko et al. (2022)	Base conceitual
Filtragem por conteúdo	Lops et al. (2011); Papadakis et al. (2023)	Núcleo do recomendador
Modelo espaço-vetorial / TF-IDF	Salton e McGill (1983); Permana e Wibowo (2023)	Vetorização do *metadata soup*
Similaridade de cosseno	Manning et al. (2008); Permana e Wibowo (2023)	Cálculo de similaridade entre itens
Sistemas híbridos	Burke (2002); Ko et al. (2022)	*Ranking* com múltiplos sinais
Recomendação sensível a contexto	Adomavicius e Tuzhilin (2005); Casillo et al. (2022)	Pesos dinâmicos de *ranking* e picks editoriais por momento
Fatoração matricial	Koren, Bell e Volinsky (2009); Isinkaye (2021)	Referência para trabalhos futuros
Cold start	Schein et al. (2002); Panda e Ray (2022)	*Onboarding* e itens populares como *fallback*
Embeddings semânticos	Reimers e Gurevych (2019); Gao et al. (2021)	Modo principal da busca por texto livre; TF-IDF como contingência

4 Dados

4.1 Fonte, Coleta e Variáveis

Os dados utilizados neste trabalho são provenientes da API pública do The Movie Database (TMDb), plataforma colaborativa que disponibiliza metadados de filmes e séries produzidos em diferentes países e idiomas. A coleta foi realizada por meio de um script de extração paginada, responsável por iterar sobre os endpoints de descoberta da API, `/discover/movie` para filmes e `/discover/tv` para séries, recuperando sucessivamente as páginas de resultados até atingir o volume desejado de registros.

O processo de coleta gerou um conjunto bruto composto por 19.998 títulos, distribuídos igualmente entre filmes e séries, com 9.999 registros de cada tipo. Os dados foram persistidos inicialmente em formato CSV para posterior processamento e utilização no sistema de recomendação.

O conjunto bruto é composto por 15 variáveis, apresentadas na Tabela 2. Entre elas, destacam-se os campos textuais *sinopse*, *generos* e *keywords*, utilizados na representação dos itens para o sistema de recomendação baseado em conteúdo. Os atributos *nota_media*, *total_votos* e *popularidade* são empregados nos mecanismos de ranqueamento, enquanto os campos *titulo*, *titulo_original*, *idioma_original* e *ano* auxiliam na identificação, indexação e enriquecimento semântico dos registros.

Tabela 2 – Variáveis do conjunto de dados bruto

Variável	**Descrição**
`id`	Identificador único do título no TMDb
`media_type`	Tipo de mídia: filme (*movie*) ou série (*tv*)
`titulo`	Título principal disponível
`titulo_original`	Título original na língua de produção
`sinopse`	Descrição textual do conteúdo
`data_lancamento`	Data de estreia ou primeiro episódio
`ano`	Ano de lançamento extraído da data
`idioma_original`	Código do idioma original de produção
`popularidade`	Indicador de popularidade fornecido pelo TMDb
`nota_media`	Avaliação média dos usuários (0 a 10)
`total_votos`	Quantidade de votos recebidos
`generos`	Lista de gêneros associados ao título
`keywords`	Palavras-chave descritivas do conteúdo
`adulto`	Indicador de conteúdo adulto
`poster_path`	Caminho da imagem de capa no servidor do TMDb

Além do catálogo de títulos, o sistema utiliza dados comportamentais produzidos

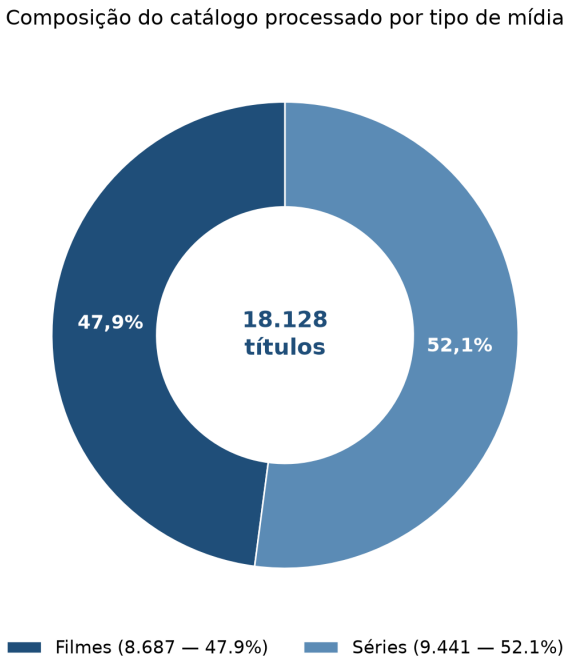
pelos próprios usuários e armazenados no banco de dados Supabase. Esses registros incluem curtidas, favoritos, histórico de assistidos, avaliações numéricas, comentários e informações de perfil. Embora não participem do treinamento ou da modelagem offline, tais dados são utilizados em tempo de execução para alimentar os mecanismos de personalização da aba *Para Você*.

A utilização dessas interações também contribui para mitigar o problema de *cold start*, caracterizado pela escassez inicial de informações sobre novos usuários ou itens (Schein *et al.*, 2002). Dessa forma, mesmo usuários recém-cadastrados podem receber recomendações personalizadas a partir das preferências informadas durante o processo de onboarding.

4.2 Análise Exploratória dos Dados

Após as etapas de limpeza, padronização e consolidação dos registros, o catálogo final utilizado pelo modelo passou a conter 18.128 títulos, sendo 8.687 filmes (47,9%) e 9.441 séries (52,1%). Além das variáveis originais, foram criados dois atributos derivados: *sinopse_len*, correspondente ao comprimento da sinopse em caracteres, e *sinopse_words*, correspondente à quantidade de palavras presentes na descrição textual.

Figura 1 – Composição do catálogo processado por tipo de mídia



Fonte: Elaborado pela autora

Os aproximadamente 1.870 registros removidos durante o pré-processamento correspondiam a entradas cuja representação textual consolidada (gêneros, palavras-chave e sinopse) resultou completamente vazia, tornando inviável a construção do *metadata_soup*

utilizado pelo sistema de recomendação. O processo de preparação dos dados também incluiu a normalização de acentuação e pontuação, a substituição de valores nulos por cadeias vazias nos campos textuais e a conversão de variáveis numéricas com tratamento de valores inválidos.

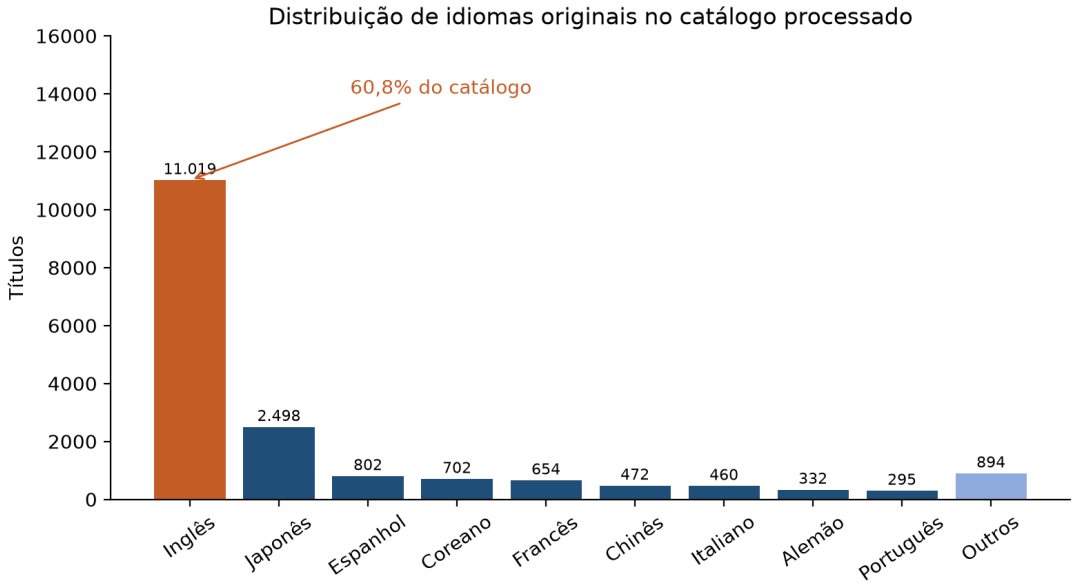
Em termos de cobertura temporal, o catálogo contempla produções de diferentes períodos históricos, com forte concentração em títulos recentes. Aproximadamente 74,9% dos registros correspondem a obras lançadas a partir do ano 2000. Os títulos a partir de 2010 concentram a maior parte do catálogo: a década de 2010–2019 responde por 32,6% dos registros (5.919 títulos), enquanto os anos de 2020 em diante correspondem a 26,3% (4.767 títulos). Essa concentração é coerente com a disponibilidade típica de catálogos de *streaming*, compostos majoritariamente por conteúdo contemporâneo, conforme sintetizado na Tabela 3.

Tabela 3 – Distribuição temporal do catálogo processado por década

Período	**Títulos**	**Percentual**
Até 1999	4.543	25,1%
2000–2009	2.888	15,9%
2010–2019	5.919	32,6%
2020–2026	4.767	26,3%
Total	18.128	100%

Quanto ao idioma original das produções, observa-se predominância do inglês, responsável por 11.019 registros (60,8%), seguido por japonês (2.498 registros), espanhol (802 registros), coreano (702 registros) e francês (654 registros). Essa diversidade linguística motivou a utilização de um modelo multilíngue de embeddings como mecanismo complementar de busca semântica.

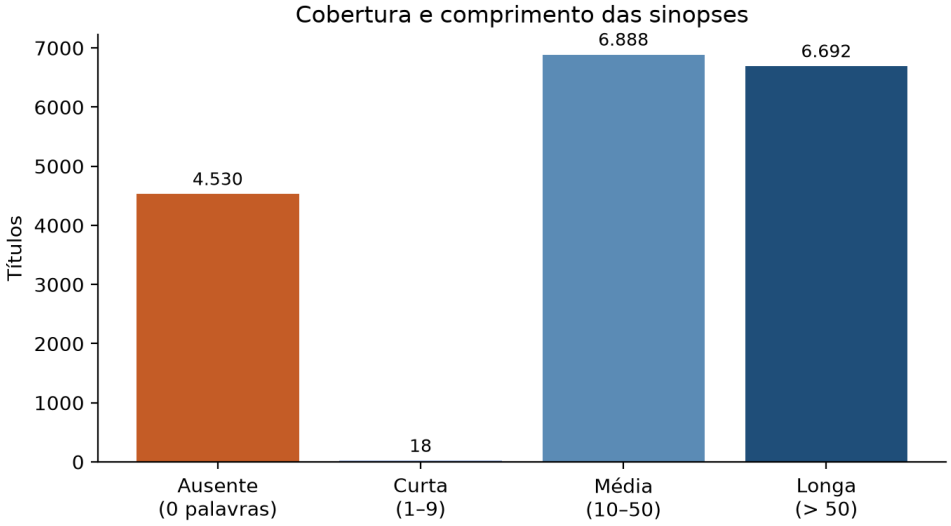
Figura 2 – Distribuição de idiomas originais no catálogo processado



Fonte: Elaborado pela autora

Entre os atributos textuais, a sinopse apresenta ausência em 4.530 registros, correspondendo a aproximadamente 25% do catálogo processado. A indisponibilidade desse campo ocorre com maior frequência em filmes (29,9%) do que em séries (20,5%). O campo de palavras-chave (*keywords*) apresenta ausência em 1.128 registros, equivalente a 6,2% do conjunto de dados.

Figura 3 – Cobertura e comprimento das sinopses



Fonte: Elaborado pela autora

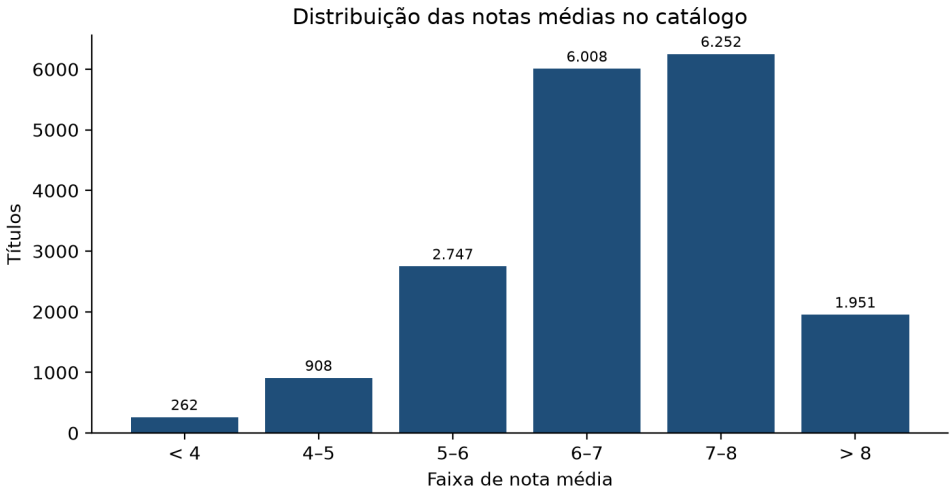
Nos casos em que a sinopse não está disponível, a representação textual utilizada pelo sistema é construída apenas a partir dos gêneros e palavras-chave existentes. Embora essa estratégia preserve a capacidade de recomendação, a ausência de descrições textuais

reduz a riqueza semântica da representação dos itens e pode impactar a qualidade das recomendações geradas.

As principais limitações dos dados estão relacionadas à dependência de uma fonte externa sujeita a atualizações e inconsistências, à coexistência de múltiplos idiomas nos campos textuais e às diferenças de popularidade entre títulos, que afetam a confiabilidade das avaliações médias. Para reduzir esse último efeito, o sistema emprega mecanismos de pontuação ponderada durante o processo de ranqueamento, reduzindo o impacto de títulos com pequeno volume de avaliações.

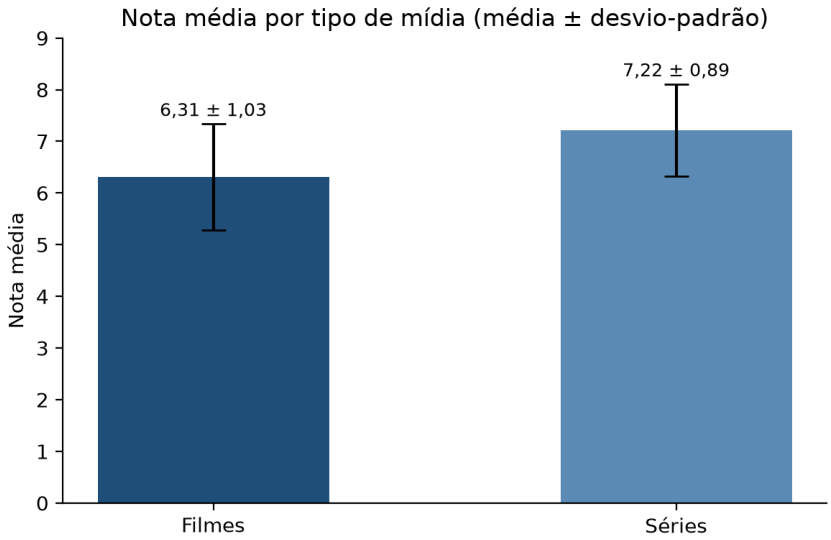
A distribuição das notas médias apresenta média de 6,79 e desvio padrão de 1,06, com mediana de 6,90, indicando uma distribuição relativamente simétrica em torno do centro. Observa-se, contudo, diferença sistemática entre os tipos de mídia: séries apresentam nota média de 7,22 (desvio de 0,89), superior à média de 6,31 (desvio de 1,03) observada para filmes, possivelmente refletindo um viés de seleção, já que produções seriadas que permanecem em exibição por mais tempo tendem a acumular avaliadores mais engajados.

Figura 4 – Distribuição das notas médias no catálogo



Fonte: Elaborado pela autora

Figura 5 – Nota média por tipo de mídia (média ± desvio-padrão)



Fonte: Elaborado pela autora

A distribuição do número de votos por título é fortemente assimétrica: a mediana é de apenas 69 votos, enquanto a média é de 887, e o quantil de 75% corresponde a apenas 343 votos, ou seja, três quartos do catálogo possuem menos de 350 avaliações. Essa característica compromete a confiabilidade da nota média isolada como critério de ranqueamento para a maior parte dos títulos, motivando diretamente a adoção da pontuação ponderada (Equação 5.3) como critério de desempate.

O campo de popularidade, calculado pelo próprio TMDb a partir de *pageviews*, votos e atividade recente, também apresenta forte assimetria positiva: média de 10,72, mediana de 8,36 e valor máximo observado de 652,61, com coeficiente de variação de 128%. Esse padrão sustenta o uso da popularidade como um sinal complementar de desempate no ranqueamento, e não como critério exclusivo de relevância, já que títulos pouco conhecidos podem apresentar alta qualidade combinada a baixa popularidade.

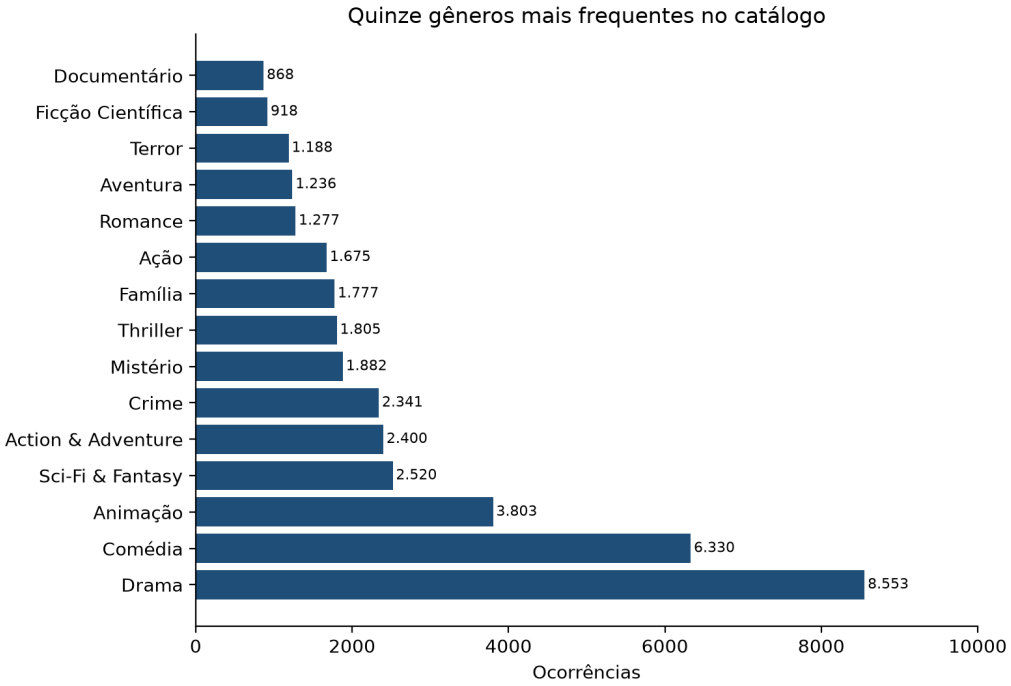
Tabela 4 – Resumo descritivo de votos e popularidade

Estatística	**Total de votos**	**Popularidade**
Média	887	10,72
Mediana	69	8,36
3º quartil (75%)	343	11,04
Máximo	39.064	652,61
Coeficiente de variação	–	128%

Em relação aos gêneros, Drama e Comédia são as categorias mais frequentes, presentes, em conjunto, em mais de 80% dos títulos do catálogo. Observa-se ainda uma inconsistência de nomenclatura entre filmes e séries herdada da própria API do TMDb, por exemplo, o gênero “Ação” é registrado separadamente de “Action & Adventure”, o

que pode fragmentar artificialmente a contagem de afinidade por gênero e representa um ponto de atenção para trabalhos futuros de normalização do catálogo.

Figura 6 – Quinze gêneros mais frequentes no catálogo



Fonte: Elaborado pela autora

Esses achados orientaram decisões específicas de modelagem, como a adoção da pontuação ponderada para mitigar o efeito de baixo volume de votos, o uso da popularidade como sinal secundário no *ranking* híbrido e a atenção à cobertura textual incompleta discutida acima.

5 Metodologia

A metodologia adotada segue um fluxo incremental de desenvolvimento de produto de dados, composto por etapas sequencialmente dependentes e orientado pelas características do catálogo apresentadas no capítulo anterior. As três seções a seguir descrevem, respectivamente, o pré-processamento e a construção da representação textual dos itens; a vetorização e o cálculo de similaridade que constituem o núcleo do recomendador; e a camada de personalização e *ranking* híbrido que gera as recomendações da aba *Para Você*. As escolhas privilegiam execução local, interpretabilidade e funcionamento com poucos dados comportamentais: TF-IDF para uma base lexical auditável, *embeddings* multilíngues para consulta livre, cosseno para comparar vetores independentemente de magnitude e regras explícitas para personalização enquanto não há logs suficientes para aprendizagem supervisionada.

5.1 Coleta, Pré-processamento e Representação Textual dos Itens

Os dados foram extraídos da API do TMDb por meio de um processo de coleta paginada, iterando sobre os *endpoints* de descoberta para filmes e séries e armazenando os resultados brutos em arquivos locais para posterior processamento. Após a coleta, foi realizada uma etapa de pré-processamento responsável pela remoção de registros sem informação de gênero, normalização de acentuação e pontuação, padronização de valores nulos como cadeias vazias e criação de variáveis auxiliares relacionadas ao tamanho das sinopses. Ao final desse processo foi obtido um catálogo composto por 18.128 títulos.

A etapa seguinte consistiu na construção da representação textual de cada item, denominada *metadata_soup*. Essa representação corresponde ao documento textual utilizado pelo sistema de recomendação baseado em conteúdo, seguindo a abordagem descrita por Lops, Gemmis e Semeraro (2011), na qual cada item é caracterizado por um conjunto de atributos descritivos relevantes. A construção do *metadata_soup*, sintetizada no Algoritmo 1, foi realizada por meio da concatenação dos campos de gêneros, palavras-chave e sinopse. Os separadores originalmente presentes nos dados foram substituídos por espaços e todo o conteúdo textual foi convertido para letras minúsculas. A ordem de concatenação foi definida de forma intencional, posicionando gêneros e palavras-chave antes da sinopse, uma vez que esses atributos representam descritores mais consistentes e menos suscetíveis a ruídos textuais.

Algoritmo 1 Construção da representação textual (*metadata_soup*)

```
1: função build_metadata_soup(item)
2:     partes ← [ ]
3:     para cada campo em [gêneros, keywords, sinopse]:
4:         txt ← normalizar(campo)
5:         txt ← texto em minúsculas
6:         substituir "|" por espaço em txt
7:         se txt não está vazio:
8:             adicionar txt a partes
9:     retornar " ".join(partes)
```

Além da representação utilizada pelo recomendador TF-IDF, foi construída uma representação textual complementar denominada *embedding_text*, composta pelos campos título, título original, gêneros, palavras-chave e sinopse. Essa estrutura é utilizada pela camada opcional de busca semântica baseada em modelos de linguagem.

5.2 Vetorização TF-IDF e Cálculo de Similaridade

Para possibilitar a comparação matemática entre os títulos do catálogo, cada item foi representado em um espaço vetorial numérico. Essa abordagem segue o modelo espaço-vetorial proposto por Salton e McGill (1983), no qual cada documento é transformado em um vetor de dimensão V, correspondente ao tamanho do vocabulário considerado. A transformação dos documentos em vetores foi realizada por meio da técnica TF-IDF (*Term Frequency–Inverse Document Frequency*), que atribui pesos aos termos considerando simultaneamente sua frequência em um documento e sua raridade em todo o corpus (Jones, 1972). O peso de cada termo é calculado conforme a Equação 5.1.

$$TFIDF(t, d, D) = freq(t, d) \times \log\left(\frac{|D|}{|\{d' \in D : t \in d'\}|}\right) \tag{5.1}$$

O fator TF representa a frequência de ocorrência do termo em determinado documento, enquanto o fator IDF reduz a influência de termos excessivamente comuns no conjunto de dados. Como consequência, palavras genéricas recebem pesos menores, enquanto termos mais específicos e discriminativos passam a exercer maior influência na representação vetorial dos títulos. Os parâmetros do `TfidfVectorizer` foram definidos considerando as características do corpus: `max_features = 10000` limita a dimensionalidade do vocabulário e reduz a presença de termos pouco representativos; `min_df = 2` elimina termos presentes em apenas um documento, reduzindo ruídos; e `ngram_range = (1,2)` permite considerar tanto unigramas quanto bigramas, preservando expressões compostas relevantes para o domínio cinematográfico.

A lista de palavras de parada foi construída manualmente para o português e versionada junto ao código-fonte, com 105 formas únicas. Partiu-se de classes funcionais

de alta frequência e baixo conteúdo temático, incluindo artigos, preposições, pronomes, conjunções, contrações e flexões recorrentes, como “a”, “de”, “para”, “que”, “uma”, “você” e “também”. As formas acentuadas foram mantidas porque o vetorizador recebe o texto antes da remoção de diacríticos. A lista foi revisada de modo conservador para não excluir termos que possam carregar informação no domínio audiovisual; nomes, gêneros e palavras temáticas não foram removidos. Na busca livre, essa lista é complementada por um conjunto separado de termos de comando, como “filme”, “série”, “quero” e “procuro”, retirados apenas da consulta por descreverem o ato de buscar, e não o conteúdo desejado. Essa construção personalizada foi necessária porque o *scikit-learn* não fornece uma lista nativa de *stopwords* em português e porque o catálogo combina metadados em mais de um idioma.

O resultado da vetorização, descrito no Algoritmo 2, é uma matriz esparsa de dimensão $N \times V$, em que N corresponde ao número de itens do catálogo e V ao número de termos selecionados. A utilização de estruturas esparsas reduz significativamente o consumo de memória, uma vez que apenas uma pequena fração dos termos está presente em cada documento.

Algoritmo 2 Vetorização TF-IDF do catálogo

```
1: função vetorizar_catalogo(df)
2:     corpus ← representação textual de cada item do catálogo
3:     inicializar vetorizador TF-IDF
4:     configurar vocabulário máximo de 10.000 termos
5:     remover termos com frequência inferior a 2 documentos
6:     considerar unigramas e bigramas
7:     aplicar lista de stopwords em português
8:     matriz_tfidf ← ajustar o vetorizador ao corpus e gerar a matriz TF-IDF
9:     retornar vetorizador e matriz_tfidf
```

Uma vez representados em formato vetorial, os títulos podem ser comparados utilizando a Similaridade de Cosseno, definida na Equação 5.2. Essa métrica mede o ângulo entre dois vetores, permitindo avaliar a proximidade temática entre documentos independentemente de seu tamanho.

$$sim(q, d_i) = \frac{q \cdot d_i}{\|q\| \times \|d_i\|} \tag{5.2}$$

Valores próximos de 1 indicam alta similaridade entre os documentos, enquanto valores próximos de 0 indicam ausência de relação significativa. A escolha dessa métrica é amplamente adotada em sistemas de recuperação de informação devido à sua invariância ao comprimento dos documentos (Manning; Raghavan; Schütze, 2008). O sistema pré-calcula e mantém em memória a matriz esparsa TF-IDF do catálogo, mas não materializa a matriz quadrada $N \times N$ de similaridades entre todos os pares. Em cada requisição, calcula-se

apenas o cosseno entre o vetor da consulta ou do título-semente e os N vetores do catálogo. Essa decisão reduz o consumo de memória de ordem quadrática para uma operação linear no número de itens por consulta e evita armazenar aproximadamente 328 milhões de pares para $N = 18.128$. A contrapartida é realizar essa multiplicação a cada busca; no catálogo estudado, a esparsidade da matriz torna o procedimento compatível com uso interativo, embora a latência não tenha sido objeto de experimento sistemático neste trabalho. Para desempates entre candidatos com escores semelhantes, foi utilizada uma adaptação da pontuação ponderada popularizada pelo IMDb, apresentada na Equação 5.3.

$$WR = \left(\frac{v}{v+m}\right) r + \left(\frac{m}{v+m}\right) c \tag{5.3}$$

em que v representa o número de votos do item, r sua avaliação média, m o limite mínimo de votos adotado e c a média global das avaliações do catálogo. O mecanismo de recomendação por título inicia pela resolução do item informado pelo usuário por meio de uma estratégia hierárquica composta por correspondência exata, correspondência por subconjunto de termos, busca parcial por *substring* e comparação aproximada utilizando *fuzzy matching*. Após a identificação do título, a similaridade é calculada entre seu vetor TF-IDF e todos os demais itens do catálogo. Exemplos práticos do *top-5* retornado pela aba Buscar para sementes conhecidas são apresentados no Capítulo de Resultados.

Já para a funcionalidade de busca textual da aba *Descobrir*, a consulta fornecida pelo usuário passa por um processo de limpeza textual, que inclui a remoção de termos genéricos sem valor discriminativo, como “filme” e “série”. O modo principal utiliza *embeddings* gerados pelo modelo *paraphrase-multilingual-MiniLM-L12-v2*[1], permitindo capturar relações semânticas mesmo quando consulta e item não compartilham os mesmos termos. Os vetores do catálogo são pré-computados offline e armazenados em cache local; a cada requisição, apenas a consulta é codificada e comparada com esse cache. Se o modelo ou o artefato não estiver disponível, o sistema recorre automaticamente ao TF-IDF. Nesse *fallback*, a consulta é transformada para o espaço vetorial lexical dos itens e recebe ainda uma ancoragem pelo termo menos frequente no corpus, que zera candidatos sem esse termo nos metadados. Portanto, os dois modos não são somados: *embeddings* são a primeira opção operacional da consulta livre e TF-IDF garante disponibilidade e comparabilidade lexical. A combinação das listas não foi adotada por falta de julgamentos de relevância para calibrar uma fusão sem favorecer indevidamente uma das escalas; comparações qualitativas entre os modos são apresentadas no Capítulo de Resultados.

[1] Variante da família Sentence-BERT (Reimers; Gurevych, 2019), otimizada para paráfrases e multilinguismo. O modelo produz vetores densos de dimensão fixa (384) e cobre mais de 50 idiomas com cerca de 118 milhões de parâmetros, o que o torna adequado ao catálogo do RecomendaAí, cujo conteúdo textual apresenta diversidade linguística, e viável para execução local, sem dependência de APIs externas e com custo computacional compatível com o pré-processamento offline e o *cache* dos embeddings.

5.3 Personalização e Ranking Híbrido

A funcionalidade *Para Você* foi desenvolvida para incorporar informações comportamentais dos usuários ao processo de recomendação. Para isso, o sistema constrói um perfil de preferências a partir das interações armazenadas no Supabase[2]. Cada interação recebe um peso proporcional ao seu nível de intencionalidade: ações de favoritar recebem peso 3,0, curtidas recebem peso 2,0, marcações como assistido recebem peso 1,5 e avaliações contribuem com o valor da nota multiplicado por 0,7. A justificativa desses pesos, fundamentada na distinção entre *feedback* implícito e explícito, é apresentada a seguir.

A definição desses valores não é arbitrária: segue uma lógica ordinal fundamentada na distinção entre *feedback* implícito e explícito discutida por Hu, Koren e Volinsky (2008) no contexto de sistemas de recomendação, segundo a qual diferentes tipos de interação carregam diferentes graus de confiança sobre a preferência real do usuário. Ações deliberadas e de baixa reversibilidade, como favoritar um título, expressam um compromisso de preferência mais forte e foram por isso associadas ao maior peso da escala (3,0). Curtidas representam uma reação positiva mais leve e reversível, recebendo peso intermediário (2,0). Marcar um título como assistido é o sinal de menor intencionalidade, indica apenas exposição ao conteúdo, não necessariamente apreciação, e por isso recebe o menor peso fixo (1,5). Já as avaliações explícitas (notas de 0 a 5) são tratadas de forma proporcional à intensidade manifestada pelo usuário (nota $\times$ 0,7), podendo inclusive superar o peso do favoritismo para notas máximas ($5 \times 0{,}7 = 3{,}5$) e aproximar-se de zero para notas muito baixas, uma vez que representam o sinal mais granular e diretamente comparável entre usuários. Essa hierarquia foi validada de forma qualitativa durante o desenvolvimento, observando-se que produzia perfis de afinidade de gênero coerentes com o histórico de navegação dos usuários de teste.

Esses pesos são utilizados para calcular uma afinidade acumulada por gênero. A partir desse perfil são selecionados títulos âncora (*seeds*) utilizados como ponto de partida para geração das recomendações, conforme resumido no Algoritmo 3. A seleção considera critérios de diversidade, limitando a quantidade de títulos pertencentes ao mesmo gênero principal e restringindo o conjunto a, no máximo, oito sementes.

[2] Plataforma *backend-as-a-service* baseada em PostgreSQL, utilizada neste trabalho como banco de dados para persistência de usuários, perfis e interações (curtidas, favoritos, assistidos, avaliações e comentários). Sua inserção na arquitetura do sistema é detalhada no Capítulo 6.

Algoritmo 3 Construção do perfil do usuário

```
1: função construir_perfil(usuario)
2:     interações ← recuperar histórico de interações do usuário
3:     afinidade_genero ← calcular afinidade acumulada por gênero
4:         ponderar cada interação conforme sua intencionalidade
5:     seeds ← selecionar itens âncora representativos
6:         garantir diversidade entre os itens selecionados
7:         limitar a seleção a no máximo 8 itens
8:     retornar seeds e afinidade_genero
```

A geração de candidatos combina três mecanismos complementares:

- recomendações obtidas por similaridade de conteúdo a partir de cada semente do perfil;
- um conjunto de títulos populares que atendem critérios mínimos de qualidade e quantidade de avaliações;
- candidatos de exploração, extraídos de gêneros secundários do perfil do usuário com peso de similaridade de conteúdo fixo e reduzido, cujo objetivo é ampliar a diversidade e a capacidade de descoberta além do núcleo estrito de preferências já manifestadas pelo usuário.

Sobre esse conjunto é aplicado um mecanismo de *ranking* híbrido inspirado em sistemas de recomendação híbridos descritos por Burke (2002), cujo escore final é definido pela Equação 5.4.

$$\begin{aligned} S_i = {} & 0{,}34 \cdot \widetilde{sim}_{i,\text{conteudo}} + 0{,}32 \cdot \widetilde{afinidade}_{i,\text{genero}} \\ & + 0{,}14 \cdot \widetilde{popularidade}_i + 0{,}06 \cdot \widetilde{recencia}_i \\ & + 0{,}14 \cdot \widetilde{frequencia}_{i,\text{seeds}} \end{aligned} \tag{5.4}$$

Antes da ponderação, cada sinal bruto $x_{i,k}$ é transformado para o intervalo $[0, 1]$ por normalização Min–Max dentro do pool de candidatos da requisição, conforme a Equação 5.5. O til na Equação 5.4 identifica essas variáveis normalizadas. Se todos os candidatos possuem o mesmo valor em um componente, atribui-se zero a esse componente, pois ele não tem poder discriminativo naquela lista. Como os pesos de cada contexto somam 1, o escore híbrido também permanece entre 0 e 1.

$$\widetilde{x}_{i,k} = \begin{cases} \dfrac{x_{i,k} - \min_j(x_{j,k})}{\max_j(x_{j,k}) - \min_j(x_{j,k})}, & \text{se } \max_j(x_{j,k}) > \min_j(x_{j,k}), \\ 0, & \text{caso contrário.} \end{cases} \tag{5.5}$$

Cada componente da Equação 5.4 é calculado a partir de atributos do catálogo e do perfil do usuário, de modo a manter a fundamentação matemática do modelo explícita e auditável. A popularidade bruta combina, em escala logarítmica, a qualidade média do título com o volume de avaliações recebidas (Equação 5.6), evitando que títulos com poucas avaliações, mas nota alta, dominem o *ranking* apenas por variação estatística.

$$popularidade = \left(\frac{nota\_media}{10}\right) \times \log_{10}(total\_votos + 10) \tag{5.6}$$

A recência decai linearmente com a idade do título, chegando a zero para produções com trinta anos ou mais (Equação 5.7), o que privilegia moderadamente conteúdos mais atuais sem excluir títulos clássicos do catálogo.

$$recencia = \max\left(0,\ 1 - \frac{ano_{atual} - ano_{lancamento}}{30}\right) \tag{5.7}$$

Por fim, a frequência entre sementes mede a proporção de títulos âncora do perfil que recuperaram aquele candidato por similaridade de conteúdo, normalizada e saturada em 1,0 a partir de três ocorrências (Equação 5.8), em que *hits* corresponde ao número de sementes distintas que geraram o candidato.

$$frequencia_{seeds} = \min\left(\frac{hits}{3},\ 1\right) \tag{5.8}$$

Os pesos apresentados na Equação 5.4 correspondem ao cenário padrão (contexto neutro) do sistema. Esses pesos, entretanto, não são fixos: uma função de resolução de contexto ajusta a ponderação de acordo com o horário e o dia da semana no fuso *America/Sao_Paulo*, ou de acordo com uma preferência de momento informada explicitamente pelo usuário na interface (Automático, Leve, Intenso ou Maratonar), com prioridade para a escolha explícita sobre a inferência temporal. Essa estratégia opera de forma alinhada ao conceito de recomendação sensível a contexto discutido por Adomavicius e Tuzhilin (2005), retomado na Revisão da Literatura. A Tabela 5 apresenta os conjuntos de pesos utilizados em cada cenário.

Tabela 5 – Pesos do ranking híbrido por contexto

Contexto	**Conteúdo**	**Gênero**	**Popularidade**	**Recência**	**Seeds**	**Critério de ativação**
Padrão	0,34	0,32	0,14	0,06	0,14	Contexto neutro
Leve	0,32	0,28	0,18	0,08	0,14	Preferência explícita
Intenso	0,38	0,34	0,10	0,04	0,14	Preferência explícita
Maratonar	0,28	0,26	0,20	0,12	0,14	Preferência explícita
Noite de lazer	0,30	0,28	0,18	0,10	0,14	Sex./sáb./dom. a partir das 18h
Manhã útil	0,36	0,34	0,12	0,04	0,14	Dias úteis, 6h–12h

A viabilidade de tornar os pesos do *ranking* sensíveis ao momento da busca foi investigada antes de sua implementação. Entre as abordagens possíveis, que vão desde modelos de *bandits* contextuais[3] e aprendizado supervisionado de pesos a partir de logs de engajamento, até heurísticas baseadas em regras de contexto, optou-se pela abordagem baseada em regras apresentada na Tabela 5, alinhada ao referencial de Adomavicius e Tuzhilin (2005) sobre recomendação sensível a contexto. Essa escolha foi motivada por três fatores: (i) a ausência de volume suficiente de interações rotuladas por contexto para treinar um modelo supervisionado, dado o estágio inicial de adoção da plataforma; (ii) a preferência por manter a interpretabilidade da personalização, coerente com os demais critérios de projeto adotados no trabalho; e (iii) a possibilidade de validar rapidamente hipóteses de uso (por exemplo, se sextas-feiras à noite favorecem conteúdo mais leve) sem depender de um período de coleta de dados anterior ao lançamento da aplicação. Como direção de pesquisa futura, essas regras poderiam ser substituídas ou complementadas por pesos aprendidos automaticamente a partir do comportamento agregado dos usuários por horário e dia da semana, à medida que a base de interações da plataforma crescer.

Os pesos foram definidos para priorizar a similaridade de conteúdo e a aderência ao perfil do usuário, mantendo influência complementar de popularidade, recência e recorrência entre os títulos geradores das recomendações, com ajustes proporcionais conforme o contexto vigente. Importa notar que a resolução de contexto altera apenas esses pesos, sem regenerar o conjunto de candidatos; o efeito empírico dessa escolha na interface é discutido no Capítulo de Resultados. Itens já consumidos ou utilizados como sementes são removidos da lista final. Após o cálculo do escore, a lista de candidatos passa por uma etapa adicional de *diversidade guiada*: aplica-se uma perturbação estocástica leve ao escore de cada candidato, com maior intensidade para gêneros de afinidade moderada no perfil do usuário, favorecendo exploração controlada em vez de aleatoriedade pura, e um limite máximo de itens por gênero principal, tipicamente entre dois e quatro títulos, dependendo do tamanho da lista solicitada. Esse mecanismo evita listas de recomendação monotemáticas, especialmente para usuários cujo histórico está concentrado em poucos gêneros. Por fim, o rótulo do contexto aplicado (por exemplo, "preferência: conteúdo leve" ou "contexto: noite de lazer") e os pesos efetivamente utilizados são retornados junto com os resultados, reforçando a interpretabilidade da personalização perante o usuário. O processo completo está detalhado no Algoritmo 4.

[3] *Contextual bandits* são algoritmos de aprendizado por reforço em que, a cada interação, o sistema observa um contexto (por exemplo, horário ou perfil do usuário), escolhe uma ação (como um conjunto de pesos de *ranking*) e recebe um retorno (*reward*) associado ao engajamento. O objetivo é equilibrar a exploração de novas configurações e o aproveitamento das que já se mostraram eficazes, aprendendo ao longo do tempo quais pesos funcionam melhor em cada contexto.

Algoritmo 4 Ranking híbrido da aba *Para Você*

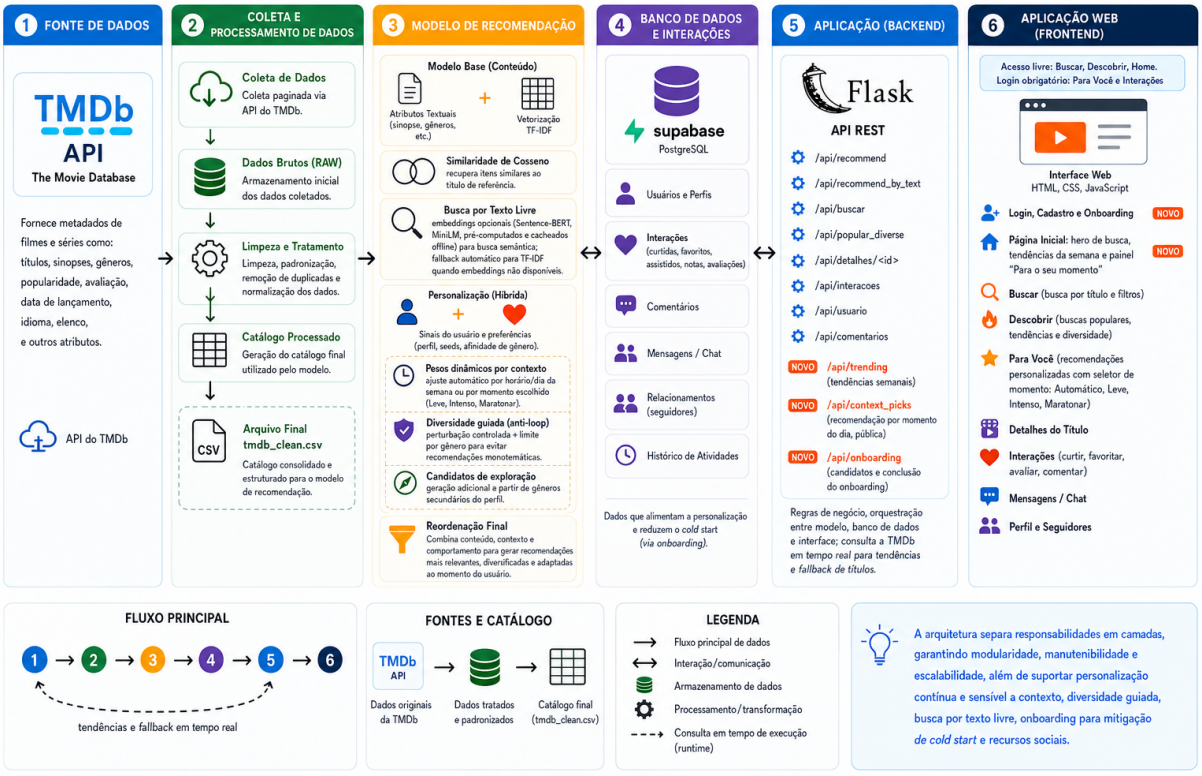
```
1: função ranking_hibrido(candidatos, perfil, top_n, momento)
2:     pesos ← resolver pesos de acordo com o momento informado pelo usuário ou com o horário atual
3:     para cada candidato em candidatos:
4:         ignorar itens já consumidos pelo usuário
5:         ignorar itens utilizados como âncoras do perfil
6:         afinidade ← calcular compatibilidade entre os gêneros do item e as preferências do usuário
7:         popularidade ← calcular relevância com base na nota média e na quantidade de avaliações
8:         recência ← calcular fator temporal com base no ano de lançamento
9:         frequência_seeds ← medir quantas âncoras do perfil contribuíram para a recuperação do item
10:    normalizar cada sinal em [0, 1] no conjunto de candidatos
11:    para cada candidato em candidatos:
12:        score ← combinar os cinco sinais normalizados, ponderados pelos pesos do contexto
13:    ordenar candidatos por score em ordem decrescente
14:    selecionados ← aplicar diversidade guiada (perturbação por afinidade de gênero + limite por gênero principal)
15:    retornar os top_n candidatos de selecionados
```

A etapa de diversidade guiada foi introduzida como resposta direta a um problema observado empiricamente durante os testes do sistema: sem esse mecanismo, o *ranking* híbrido tendia a reforçar excessivamente um único gênero ou subtema do histórico do usuário, por exemplo, concentrar-se quase exclusivamente em filmes de super-heróis para usuários que haviam interagido com poucos títulos desse gênero, um efeito de aprisionamento temático análogo ao observado em *feeds* de recomendação por engajamento, como os de plataformas de vídeos curtos. Esse comportamento é uma manifestação prática da tendência à superespecialização apontada por Lops, Gemmis e Semeraro (2011) como limitação típica de sistemas de filtragem baseada em conteúdo, já discutida na Revisão da Literatura. A diversidade guiada consolida-se, portanto, como funcionalidade do sistema, e não apenas como ajuste pontual, buscando mitigar esse efeito sem abandonar a personalização, ao equilibrar fidelidade ao perfil do usuário com exploração controlada de gêneros adjacentes.

6 Arquitetura do Sistema

A arquitetura do sistema RecomendaAí foi estruturada em camadas, com o objetivo de separar responsabilidades, facilitar a manutenção e permitir evolução modular da solução. A Figura 7 apresenta uma visão geral dos componentes e do fluxo de dados da aplicação.

Figura 7 – Arquitetura geral do sistema RecomendaAí



Fonte: Elaborado pela autora

O fluxo do sistema inicia-se na camada de dados, responsável pela coleta de informações provenientes da API do TMDb. Os dados obtidos são armazenados inicialmente em formato bruto e posteriormente submetidos a etapas de limpeza, padronização e seleção de atributos relevantes. Em tempo de execução, essa camada também é responsável por consultar a API do TMDb para obter tendências semanais e por realizar consultas de *fallback* quando um título buscado pelo usuário não está presente no catálogo processado localmente.

Na camada de modelagem, o sistema utiliza uma abordagem baseada em conteúdo, na qual os atributos textuais são transformados em representações vetoriais utilizando TF-IDF (Salton; McGill, 1983; Jones, 1972). A similaridade entre os itens é calculada por

meio da métrica de cosseno (Manning; Raghavan; Schütze, 2008), permitindo recuperar conteúdos semanticamente próximos. Adicionalmente, aplica-se uma etapa de reordenação híbrida baseada em sinais comportamentais dos usuários (Burke, 2002), que incorpora pesos sensíveis ao contexto temporal e à preferência explícita do usuário, além de um mecanismo de diversidade guiada aplicado à lista final de recomendações, conforme detalhado no Capítulo de Metodologia.

A camada de aplicação é implementada em Flask e é responsável pela orquestração das requisições, integração entre modelo e persistência, além da disponibilização de endpoints para funcionalidades de busca, exploração de catálogo e geração de recomendações.

Na camada de persistência, o sistema utiliza o Supabase (PostgreSQL) para armazenamento de dados estruturados, incluindo usuários, perfis, interações, avaliações e relacionamentos. Esses dados sustentam os mecanismos de personalização contínua e mitigação de cold start.

Por fim, a camada de apresentação consiste em uma interface web desenvolvida com HTML, CSS e JavaScript, responsável por consumir os serviços do backend e apresentar os resultados ao usuário de forma interativa e organizada. Parte dessa camada é acessível também a visitantes não autenticados, conforme detalhado no capítulo seguinte.

7 Resultados

O principal resultado deste trabalho é a disponibilização de uma aplicação web funcional de recomendação de filmes e séries, denominada RecomendaAí, acessível em `https://huggingface.co/spaces/lorenaxmn/recomendaai`. Os resultados são apresentados sob duas perspectivas complementares: a perspectiva técnica, correspondente à construção do fluxo completo de Ciência de Dados, coleta, pré-processamento, análise exploratória, vetorização textual, *ranking* híbrido sensível a contexto e persistência de interações; e a perspectiva de produto, correspondente à entrega dessas recomendações ao usuário final por meio de uma interface web organizada em áreas de busca, exploração e personalização.

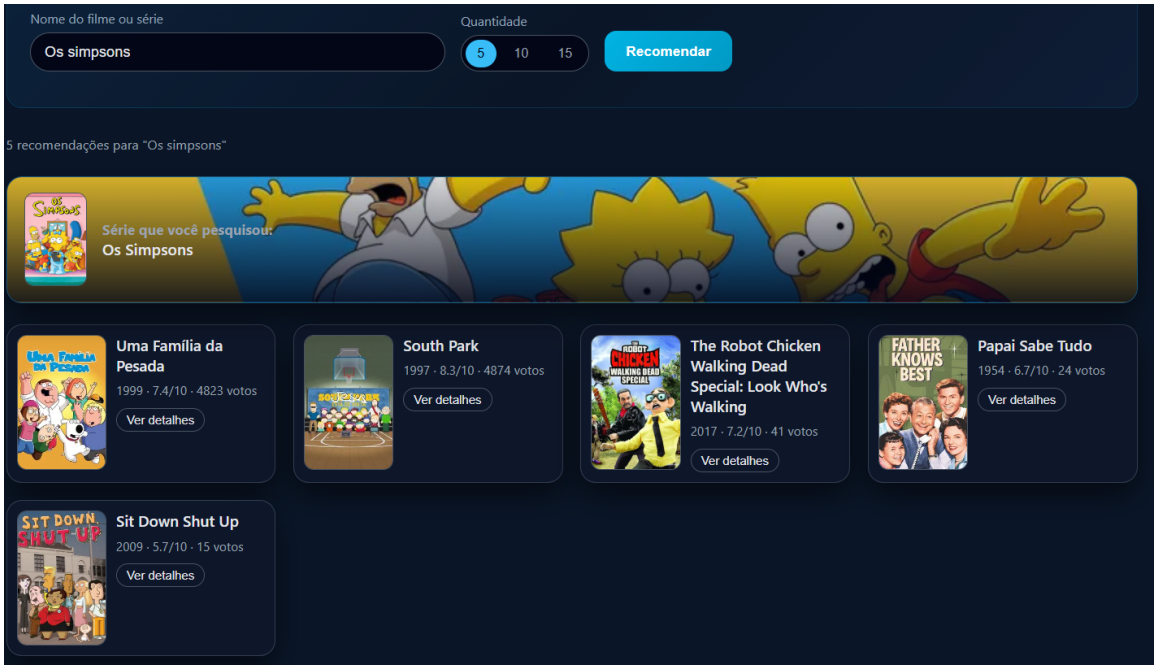
Ao acessar o RecomendaAí, o usuário chega diretamente à página principal da plataforma, sem necessidade de criar conta ou realizar login. Esse modo aberto reúne, desde o primeiro acesso, a aba Buscar, a aba Descobrir, a página de detalhe dos títulos, um painel de tendências da semana e um painel editorial denominado "Para o seu momento", que seleciona títulos do catálogo de acordo com o dia e o horário atuais (por exemplo, thrillers e mistérios na madrugada, comédias no horário de almoço em dias úteis, ou títulos de ação e aventura recentes na noite de sexta-feira). Esse painel opera de forma independente de qualquer histórico de interações do usuário e ilustra, na prática, o conceito de recomendação sensível a contexto discutido na Revisão da Literatura (Adomavicius; Tuzhilin, 2005). Essa escolha de projeto busca reduzir a barreira de entrada, permitindo que qualquer visitante experimente o valor do sistema, buscando um título conhecido ou descrevendo em linguagem natural o que deseja assistir, antes de decidir se quer criar uma conta.

A aba Buscar permite localizar recomendações semelhantes a partir de um filme ou série informado pelo usuário. O sistema utiliza vetorização TF-IDF e cálculo de similaridade de cosseno para identificar conteúdos semanticamente próximos ao item selecionado; quando o título informado não é encontrado no catálogo processado localmente, o sistema realiza uma consulta de *fallback* diretamente à API do TMDb, ampliando a cobertura de títulos passíveis de recomendação. Já a aba Descobrir oferece formas mais amplas de exploração do catálogo, permitindo buscas em linguagem natural, filtragem por gêneros e utilização de títulos de referência para geração das recomendações. Essas modalidades podem ser utilizadas individualmente ou de forma combinada, ampliando as possibilidades de descoberta de conteúdos, e estão disponíveis tanto para visitantes quanto para usuários autenticados.

Para ilustrar o comportamento prático da aba Buscar, foram selecionados três títulos-semente e registrados o *top-5* retornado com o respectivo escore de similaridade de

cosseno. Os resultados, ilustrados nas Figuras 8, 9 e 10 e quantificados nas Tabelas 6, 7 e 8, mostram coerência temática com as sementes: animações adultas e sitcoms familiares a partir de *Os Simpsons*; títulos do universo Holmes a partir de *Enola Holmes*; e produções ligadas ao Superman a partir de *Smallville*.

Figura 8 – Recomendações na aba Buscar para a semente *Os Simpsons*

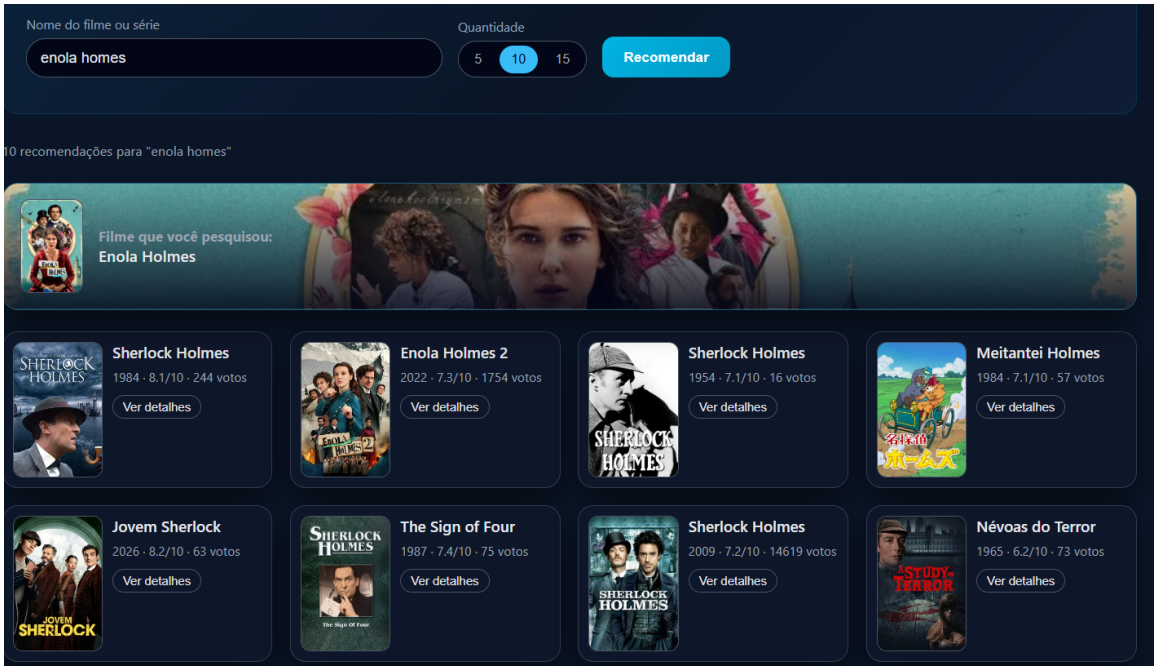


Fonte: Elaborado pela autora

Tabela 6 – Top-5 recomendações para a semente *Os Simpsons* (TF-IDF + cosseno)

Pos.	**Título recomendado**	**Ano**	**Similaridade**
1	Uma Família da Pesada	1999	0,384
2	South Park	1997	0,315
3	The Robot Chicken Walking Dead Special	2017	0,310
4	Papai Sabe Tudo	1954	0,307
5	Sit Down Shut Up	2009	0,270

Figura 9 – Recomendações na aba Buscar para a semente *Enola Holmes*

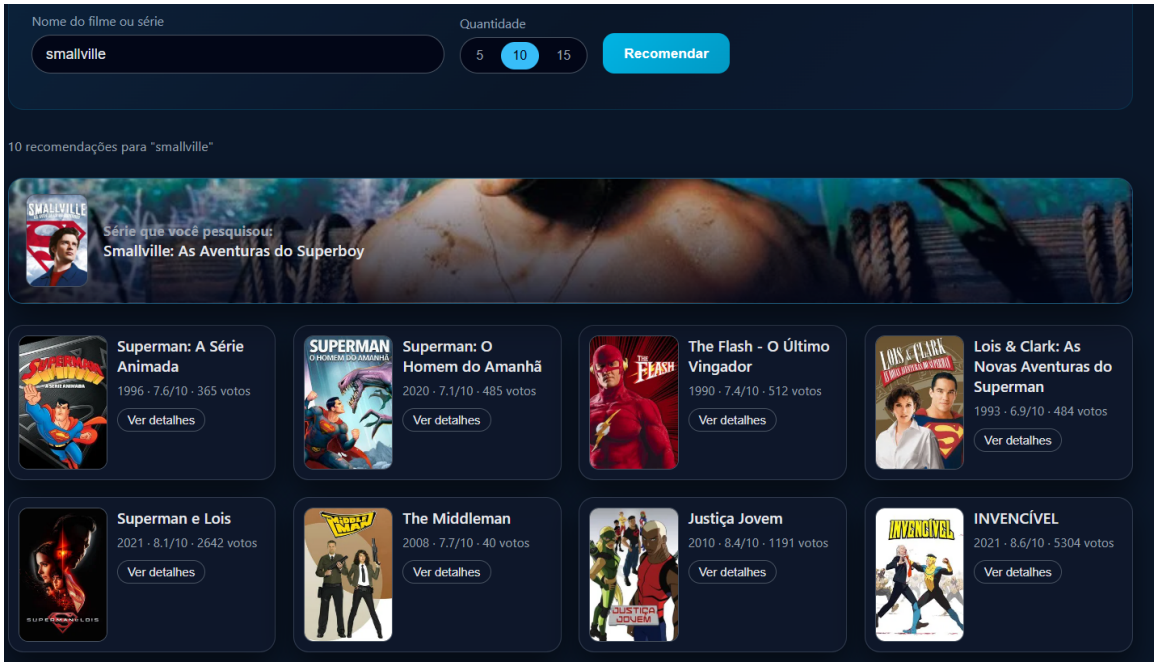


Fonte: Elaborado pela autora

Tabela 7 – Top-5 recomendações para a semente *Enola Holmes* (TF-IDF + cosseno)

Pos.	Título recomendado	Ano	Similaridade
1	Sherlock Holmes	1984	0,425
2	Enola Holmes 2	2022	0,404
3	Sherlock Holmes	1954	0,403
4	Jovem Sherlock	2026	0,370
5	Meitantei Holmes	1984	0,366

Figura 10 – Recomendações na aba Buscar para a semente *Smallville*



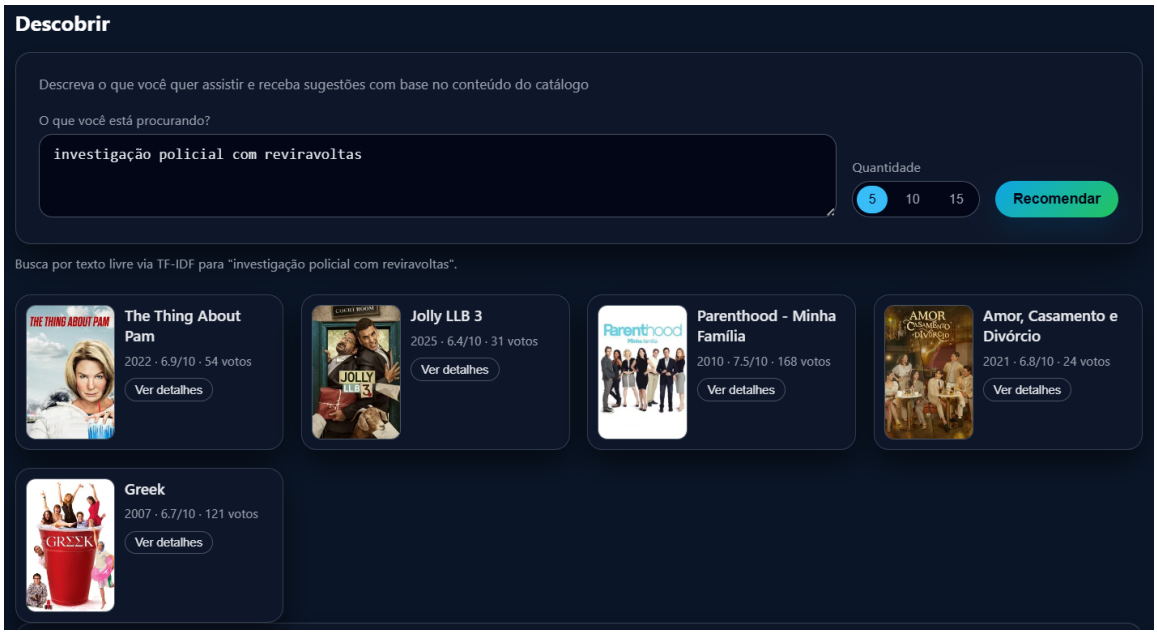
Fonte: Elaborado pela autora

Tabela 8 – Top-5 recomendações para a semente *Smallville* (TF-IDF + cosseno)

Pos.	Título recomendado	Ano	Similaridade
1	Superman: A Série Animada	1996	0,332
2	Superman: O Homem do Amanhã	2020	0,309
3	The Flash – O Último Vingador	1990	0,306
4	Lois & Clark: As Novas Aventuras do Superman	1993	0,302
5	Superman e Lois	2021	0,279

Na aba Descobrir, foram executadas três consultas em linguagem natural: "investigação policial com reviravoltas", "ficção científica sobre viagem no tempo" e "tubarão gigante". As Figuras 11, 12 e 13 registram a interface no modo *fallback* TF-IDF, e as Tabelas 9, 10 e 11 comparam o *top-5* devolvido por esse mecanismo com o modo principal baseado em *embeddings* semânticos. Os escores de cosseno devem ser comparados apenas dentro de cada coluna, pois espaços vetoriais lexicais e densos possuem distribuições distintas. Observa-se que o TF-IDF favorece coincidência lexical, enquanto os *embeddings* recuperam obras tematicamente alinhadas mesmo quando o vocabulário da sinopse difere da consulta. Isso não elimina erros: os resultados abaixo mostram que ambos os modos dependem da qualidade dos metadados e que proximidade vetorial não equivale necessariamente a relevância percebida.

Figura 11 – Aba Descobrir (TF-IDF) para a consulta “investigação policial com reviravoltas”



Fonte: Elaborado pela autora

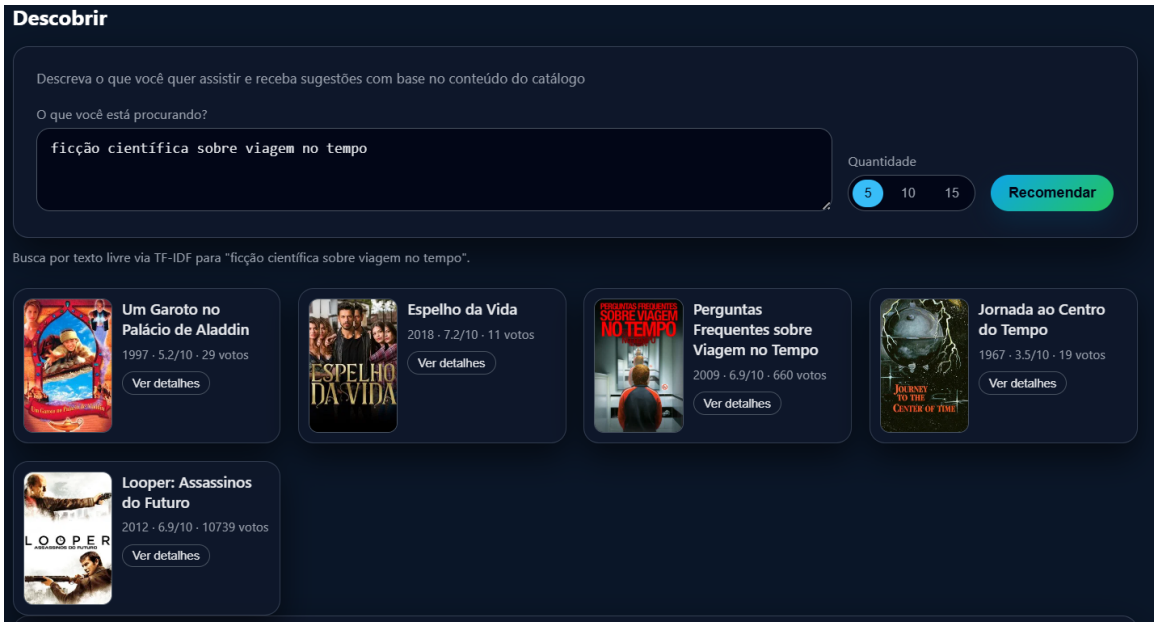
Tabela 9 – Top-5 da consulta “investigação policial com reviravoltas”: TF-IDF versus *embeddings*

Pos.	TF-IDF	Sim.	Embeddings	Sim.
1	The Thing About Pam	0,209	Trident	0,767
2	Jolly LLB 3	0,198	Polizeiruf 110	0,758
3	Parenthood - Minha Família	0,196	Kanıt	0,717
4	Amor, Casamento e Divórcio	0,180	Mentes Criminosas: Conduta Suspeita	0,717
5	Greek	0,167	MIU404	0,703

Nota: *Trident* é o nome internacional da série chinesa retornada em primeiro lugar pelos *embeddings*.

Na Tabela 9, os *embeddings* apresentam maior coerência de conjunto ao recuperar séries policiais como *Polizeiruf 110*, *Kanıt*, *Mentes Criminosas: Conduta Suspeita* e *MIU404*. O primeiro colocado, *Trident*, também pertence ao gênero policial, embora o título internacional isolado torne essa relação pouco evidente na tabela. Já o TF-IDF retorna *Parenthood – Minha Família*, *Amor, Casamento e Divórcio* e *Greek*, títulos sem relação central aparente com investigação policial. A inspeção dos metadados indica que a sobreposição ocorre por termos genéricos presentes em sinopses extensas e pela incapacidade do modelo lexical de representar a expressão composta “investigação policial com reviravoltas” como intenção única. Esses casos constituem falsos positivos qualitativos e evidenciam que a ancoragem pelo termo raro reduz, mas não elimina, associações espúrias.

Figura 12 – Aba Descobrir (TF-IDF) para a consulta “ficção científica sobre viagem no tempo”

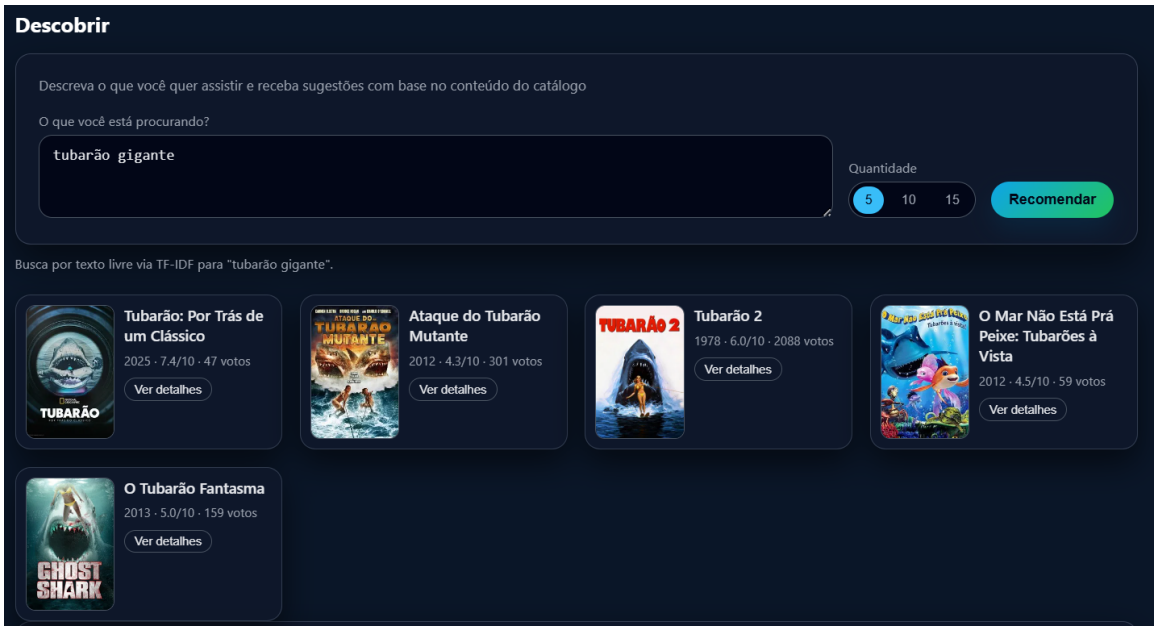


Fonte: Elaborado pela autora

Tabela 10 – Top-5 da consulta “ficção científica sobre viagem no tempo”: TF-IDF versus *embeddings*

Pos.	TF-IDF	Sim.	Embeddings	Sim.
1	Um Garoto no Palácio de Aladdin	0,414	Projeto Almanaque	0,803
2	Perguntas Frequentes sobre Viagem no Tempo	0,327	Journeyman	0,799
3	Espelho da Vida	0,320	Timecop 2 - O Guardião Do Tempo	0,786
4	Jornada ao Centro do Tempo	0,270	Turma da Mônica em Uma Aventura no Tempo	0,783
5	Looper: Assassinos do Futuro	0,256	MIB - Homens de Preto III	0,770

Figura 13 – Aba Descobrir (TF-IDF) para a consulta “tubarão gigante”



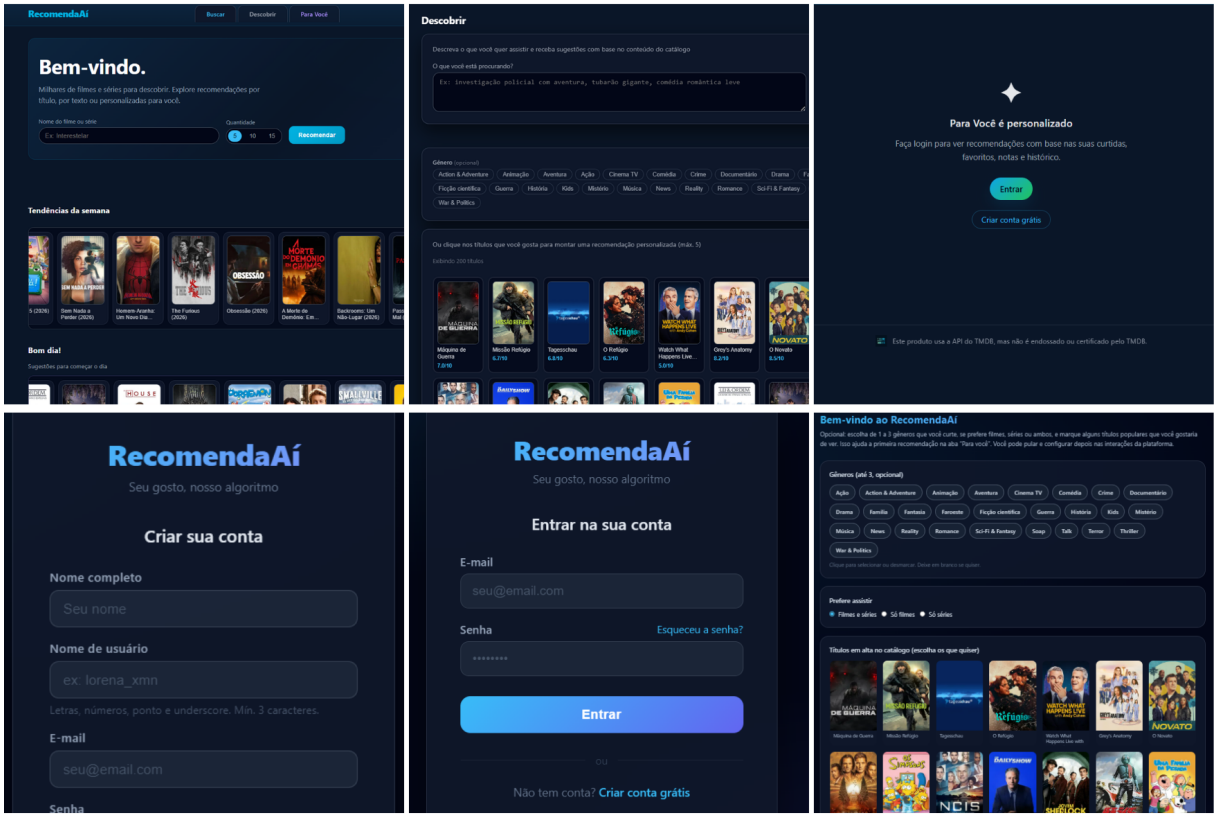
Fonte: Elaborado pela autora

Tabela 11 – Top-5 da consulta “tubarão gigante”: TF-IDF versus *embeddings*

Pos.	**TF-IDF**	**Sim.**	**Embeddings**	**Sim.**
1	Tubarão: Por Trás de um Clássico	0,360	Megatubarão 2	0,702
2	Ataque do Tubarão Mutante	0,292	Megatubarão	0,691
3	Tubarão 2	0,232	Ataque do Tubarão Mutante	0,685
4	O Mar Não Está Prá Peixe: Tubarões à Vista	0,205	Tubarão 2	0,684
5	O Tubarão Fantasma	0,153	Shark Week	0,625

A criação de conta e o login só se tornam necessários quando o usuário deseja acessar a aba Para Você, que concentra as recomendações personalizadas calculadas a partir do próprio histórico de interações dentro do aplicativo, ou registrar interações explícitas (curtidas, favoritos, avaliações e comentários) e utilizar recursos sociais (perfil e chat). A tela de login permite que o usuário autentique sua conta utilizando e-mail e senha, com suporte para recuperação de acesso e redirecionamento para criação de conta. Já a tela de cadastro solicita informações básicas, como nome, nome de usuário, e-mail e senha, realizando validações para garantir a unicidade do *username* e a consistência dos dados informados. Após o primeiro acesso, o usuário é direcionado para a etapa de onboarding, na qual pode selecionar gêneros de preferência, definir interesse entre filmes, séries ou ambos, além de marcar títulos populares como favoritos iniciais. Essas interações são utilizadas para alimentar imediatamente o mecanismo de recomendação personalizada, reduzindo o problema de *cold start* (Schein *et al.*, 2002). A Figura 14 ilustra esse fluxo de autenticação, cadastro e onboarding.

Figura 14 – Telas de login, cadastro e onboarding do RecomendaAí



Fonte: Elaborado pela autora

Uma vez autenticado, o usuário passa a ter acesso, além das abas Buscar e Descobrir e da página de detalhe dos títulos, já disponíveis desde o primeiro acesso, à aba Para Você e aos recursos de interação. A aba Para Você concentra as recomendações personalizadas da plataforma, calculadas a partir do histórico de interações do usuário, como curtidas, favoritos, avaliações e conteúdos assistidos. O ranqueamento combina critérios de similaridade textual, afinidade de gêneros, popularidade, recência e frequência entre as sementes de recomendação, produzindo uma lista híbrida de recomendações ajustada ao perfil individual do usuário. Essa aba também permite que o usuário selecione explicitamente um "momento" (Automático, Leve, Intenso ou Maratonar), ajustando os pesos do *ranking* híbrido conforme descrito no Capítulo de Metodologia. Na resposta da API, o sistema expõe o rótulo do contexto aplicado e os pesos utilizados, o que favorece a interpretabilidade técnica da personalização.

Na prática observada, a troca de momento altera sobretudo a ordem dos títulos já presentes no conjunto de candidatos gerado a partir do perfil, e não a composição desse conjunto. Isso ocorre porque os modos compartilham as mesmas sementes e o mesmo pool de candidatos, diferindo apenas nas ponderações da Equação 5.4. Para facilitar a comparação após a padronização das variáveis, a Tabela 12 apresenta o *top-5*

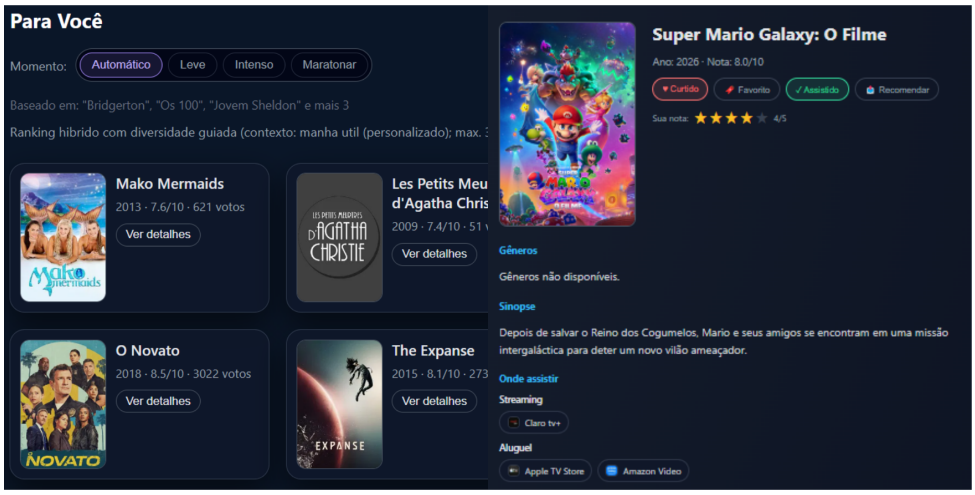
e o respectivo escore híbrido nos modos Automático (contexto padrão), Leve, Intenso e Maratonar, para um perfil ilustrativo de comédias e *sitcoms*, formado pelas sementes *Os Simpsons*, *Friends*, *The Office*, *Brooklyn Nine-Nine* e *How I Met Your Mother*. O cálculo utilizou o mesmo pool de 370 candidatos e isolou o efeito dos pesos, sem a perturbação estocástica da diversidade guiada. Os escores agora pertencem ao intervalo $[0, 1]$ e podem ser interpretados na escala comum definida pela Equação 5.5. Há elevada sobreposição, mas não identidade: Automático e Leve compartilham nove dos dez primeiros títulos (Jaccard de 0,818), enquanto Intenso e Maratonar compartilham sete (Jaccard de 0,538). No *top-5*, *The Cosby Show* e *Archie Bunker's Place* permanecem nas duas primeiras posições em todos os modos; as mudanças mais visíveis ocorrem entre *South Park*, *Desus & Mero* e *Uma Família da Pesada*.

Tabela 12 – *Top-5* da aba Para Você por modo de momento, com escore híbrido

Pos.	**Modo**	**Título recomendado**	**Escore**
1	Automático	The Cosby Show	0,7414
2	Automático	Archie Bunker's Place	0,7294
3	Automático	South Park	0,6637
4	Automático	Desus & Mero	0,6490
5	Automático	Uma Família da Pesada	0,6467
1	Leve	The Cosby Show	0,7057
2	Leve	Archie Bunker's Place	0,6867
3	Leve	South Park	0,6507
4	Leve	Uma Família da Pesada	0,6286
5	Leve	Desus & Mero	0,6223
1	Intenso	The Cosby Show	0,7785
2	Intenso	Archie Bunker's Place	0,7777
3	Intenso	Desus & Mero	0,6773
4	Intenso	South Park	0,6704
5	Intenso	Mad Bull 34	0,6661
1	Maratonar	The Cosby Show	0,6601
2	Maratonar	Archie Bunker's Place	0,6354
3	Maratonar	South Park	0,6289
4	Maratonar	Anora	0,6122
5	Maratonar	Desus & Mero	0,6080

Complementando esse fluxo, a página de detalhe do título centraliza informações específicas de cada obra, incluindo sinopse, gêneros, avaliação média, trailer e plataformas de streaming disponíveis. Nessa página também são registradas interações explícitas do usuário, como curtidas, favoritos, marcação de assistido, avaliações e comentários públicos, dados que posteriormente influenciam os próximos cálculos de recomendação, registro que exige autenticação, ainda que a visualização da página de detalhe esteja disponível também para visitantes. A Figura 15 apresenta a aba Para Você e a página de detalhe do título.

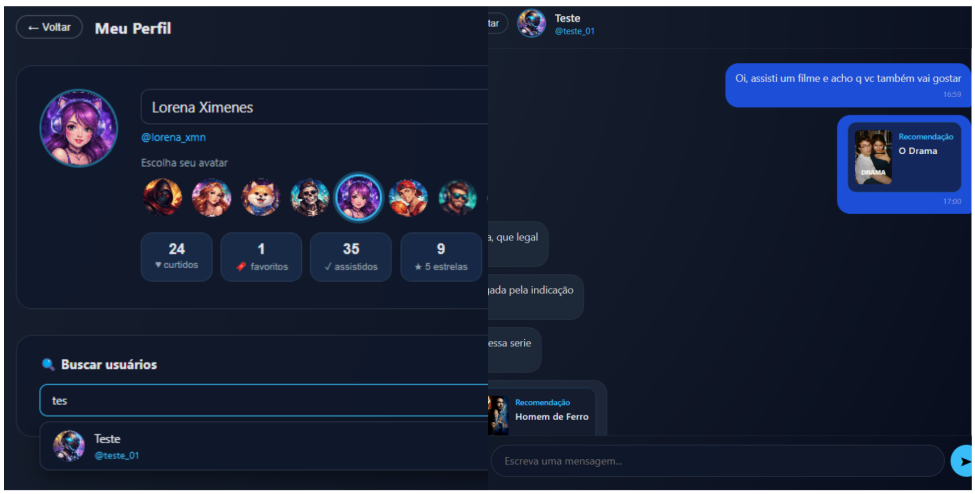
Figura 15 – Aba Para Você e página de detalhe do título



Fonte: Elaborado pela autora

Além das funcionalidades de recomendação, a plataforma também incorpora recursos sociais por meio da página de perfil e do sistema de chat entre usuários. A página de perfil reúne informações gerais da conta, como nome, avatar, quantidade de curtidas, favoritos, títulos assistidos, seguidores e usuários seguidos. O usuário também pode editar informações básicas e pesquisar outros perfis na plataforma. Já o módulo de chat permite a troca de mensagens diretas entre usuários, incluindo o compartilhamento de títulos do catálogo em formato de cartão interativo. Esses mecanismos complementam a experiência da aplicação ao estimular recomendações sociais entre usuários, ampliando o engajamento e as possibilidades de descoberta de conteúdos, conforme ilustrado na Figura 16.

Figura 16 – Página de perfil e chat entre usuários



Fonte: Elaborado pela autora

8 Considerações Finais

Este trabalho desenvolveu e disponibilizou o RecomendaAí, uma aplicação web de recomendação de filmes e séries fundamentada em filtragem baseada em conteúdo e em sinais comportamentais do usuário. O fluxo implementado contempla a coleta de dados pela API do TMDb, o pré-processamento e a representação textual dos itens, a vetorização TF-IDF com similaridade de cosseno, a busca semântica por *embeddings* com contingência lexical, o *ranking* híbrido com sinais normalizados e pesos sensíveis a contexto e a integração dessas capacidades em uma interface web com modo aberto para visitantes e personalização contínua para usuários autenticados.

Em relação aos sete objetivos específicos, o trabalho: (i) coletou e caracterizou o catálogo do TMDb, explicitando limitações de cobertura e completude; (ii) preparou representações textuais a partir de gêneros, palavras-chave e sinopses; (iii) implementou a recomendação por título com TF-IDF e similaridade de cosseno; (iv) implementou a busca semântica por *embeddings* e a comparou qualitativamente com o *fallback* TF-IDF; (v) desenvolveu um *ranking* híbrido com sinais normalizados e pesos sensíveis ao contexto; (vi) integrou essas técnicas à aplicação web, com persistência de interações, *onboarding* e contingência por itens populares; e (vii) avaliou o comportamento do protótipo por consultas e perfis ilustrativos, identificando coerência temática, falsos positivos e predominância de reordenação nos modos de momento. A escolha por um modelo interpretável, em que pesos, contextos e atributos textuais podem ser inspecionados, priorizou a transparência acadêmica e a clareza para o usuário final, em coerência com a fundamentação apresentada na Revisão da Literatura.

Retomando a justificativa do trabalho, o RecomendaAí responde diretamente ao problema da sobrecarga informacional e do paradoxo da escolha em catálogos audiovisuais extensos (Schwartz, 2004; Meza; D'Urso, 2024). Em vez de expor o usuário aos mais de 18 mil títulos do catálogo processado de forma indiferenciada, a aplicação reduz o espaço de busca a listas curtas e ranqueadas: a aba Buscar parte de um título conhecido para sugerir conteúdos semelhantes; a aba Descobrir traduz intenções em linguagem natural em um conjunto restrito de candidatos; e a aba Para Você organiza sugestões a partir do histórico de interações do usuário. Mecanismos adicionais, como o painel editorial "Para o seu momento", a diversidade guiada na lista personalizada e o acesso aberto às abas de exploração sem exigência imediata de login, reforçam essa filtragem ao diminuir o esforço cognitivo da escolha e ao apoiar a descoberta de itens relevantes, alinhando-se ao papel clássico dos sistemas de recomendação como infraestrutura de filtragem de informação (Resnick; Varian, 1997; Ricci; Rokach; Shapira, 2015).

Apesar dos resultados obtidos, o trabalho apresenta limitações que devem ser

consideradas na interpretação do sistema. A qualidade das recomendações depende da completude e da consistência dos metadados fornecidos pelo TMDb, incluindo variações de idioma e inconsistências de nomenclatura de gêneros entre filmes e séries. Em particular, o sistema herda vieses estruturais do catálogo processado: 60,8% dos títulos têm inglês como idioma original, aproximadamente 25% não possuem sinopse e 74,9% das obras foram lançadas a partir do ano 2000. Esses desequilíbrios podem favorecer conteúdos anglófonos e contemporâneos nas recomendações e reduzir a riqueza semântica dos itens sem descrição textual. O TF-IDF, embora interpretável, possui limitações no tratamento de sinônimos e de intenções subjetivas expressas em linguagem natural (Manning; Raghavan; Schütze, 2008); os *embeddings* atenuam parte desse problema, porém introduzem dependência adicional de modelo e de *cache* pré-gerado. A avaliação qualitativa também revelou falsos positivos nos dois modos, reforçando que escores altos de similaridade vetorial não substituem validação de relevância com usuários.

Usuários novos ainda podem enfrentar o problema de *cold start*, ainda que o onboarding e o uso de títulos populares reduzam seus efeitos (Schein *et al.*, 2002). Os pesos contextuais e os slots editoriais da página inicial foram definidos por heurísticas calibradas manualmente, sem otimização automática a partir de *feedback* online. Em particular, o seletor de momento da aba Para Você atua apenas sobre os pesos do *ranking* híbrido, sem filtrar nem regenerar o conjunto de candidatos: o efeito empírico observado é predominantemente de reordenação, com elevada sobreposição entre as listas de Leve, Intenso e Maratonar. A diversidade guiada favorece a descoberta e reduz o aprisionamento temático, mas introduz variabilidade entre requisições, o que dificulta a reprodução bit a bit dos resultados. Por fim, a avaliação do sistema permanece predominantemente qualitativa; estudos com usuários reais e métricas de produto ainda podem ser ampliados (Adomavicius; Tuzhilin, 2005).

Como desdobramentos futuros, destacam-se: (i) a incorporação de técnicas colaborativas para complementar o modelo baseado em conteúdo (Koren; Bell; Volinsky, 2009); (ii) a inclusão, na interface, de explicações do tipo "recomendado porque...", aproveitando o rótulo de contexto já exposto pela API; (iii) a automação da atualização periódica do catálogo e dos *embeddings*; (iv) a mensuração de métricas de uso, como cliques, salvamentos, avaliações e taxa de conversão em interações; (v) o fortalecimento do seletor de momento, por meio de filtros por tom ou gênero, de pools de candidatos distintos por contexto e/ou de pesos aprendidos (ou validados em teste A/B) a partir de dados de engajamento; (vi) a experimentação com diferentes modelos de *embeddings* e estratégias de *ranking* (Reimers; Gurevych, 2019); e (vii) a normalização das nomenclaturas de gêneros entre filmes e séries no catálogo processado.

Para além do domínio de filmes e séries, o pipeline adotado neste trabalho, coleta e limpeza de metadados textuais, representação por TF-IDF (com *embeddings* opcionais), similaridade de cosseno, personalização a partir de interações e *ranking* híbrido sensível a

contexto, é transferível a outros cenários de recomendação baseada em conteúdo. Exemplos naturais incluem sugestão de restaurantes a partir de descrições de culinária, ambiente e avaliações; recomendação de peças em lojas de roupa a partir de categorias, materiais e descrições de produto; e, de modo análogo, catálogos de livros, cursos, eventos ou serviços locais. Em cada caso, a adaptação consistiria sobretudo na substituição da fonte de dados e no ajuste dos atributos textuais e dos pesos de contexto ao domínio, preservando a estrutura metodológica e a interpretabilidade do modelo.

Em síntese, o RecomendaAí demonstra a viabilidade de integrar, em um único fluxo aplicado, conceitos de Ciência de Dados e de sistemas de recomendação, entregando uma solução utilizável, interpretável e sensível ao contexto de uso, ao mesmo tempo em que deixa abertas frentes claras de evolução metodológica e de produto.

REFERÊNCIAS

ADOMAVICIUS, Gediminas; TUZHILIN, Alexander. 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**, v. 17, n. 6, p. 734–749, 2005. DOI: 10.1109/TKDE.2005.99. Citado nas pp. 14, 31, 32, 36, 47.

AGÊNCIA NACIONAL DO CINEMA. **Panorama do mercado de vídeo por demanda no Brasil – 2022**. Rio de Janeiro, 2023. Disponível em: https://www.gov.br/ancine/pt-br/oca/publicacoes/arquivos.pdf/informe-vod2022.pdf. Acesso em: 5 ago. 2026. Citado na p. 10.

BURKE, Robin. Hybrid Recommender Systems: Survey and Experiments. **User Modeling and User-Adapted Interaction**, v. 12, n. 4, p. 331–370, 2002. DOI: 10.1023/A:1021240730564. Citado nas pp. 16, 30, 35.

CASILLO, Mario *et al.* Context Aware Recommender Systems: A Novel Approach Based on Matrix Factorization and Contextual Bias. **Electronics**, v. 11, n. 7, p. 1003, 2022. DOI: 10.3390/electronics11071003. Citado na p. 14.

COMSCORE. **Estudo customizado TV Conectada no Brasil: 4ª edição**. Reston, 2026. Disponível em: https://www.comscore.com/por/Insights/Apresentacoes-e-documentos/2026/Estudo-Customizado-TV-Conectada-no-Brasil-4a-Edicao. Acesso em: 5 ago. 2026. Citado na p. 10.

GAO, Tianyu; YAO, Xingcheng; CHEN, Danqi. SimCSE: Simple Contrastive Learning of Sentence Embeddings. *In:* PROCEEDINGS of the 2021 Conference on Empirical Methods in Natural Language Processing. [*S. l.*]: Association for Computational Linguistics, 2021. p. 6894–6910. DOI: 10.18653/v1/2021.emnlp-main.552. Citado nas pp. 15, 17.

HU, Yifan; KOREN, Yehuda; VOLINSKY, Chris. Collaborative Filtering for Implicit Feedback Datasets. *In:* 2008 Eighth IEEE International Conference on Data Mining. [*S. l.*]: IEEE, 2008. p. 263–272. DOI: 10.1109/ICDM.2008.22. Citado na p. 29.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Pesquisa Nacional por Amostra de Domicílios Contínua: acesso à internet e à televisão e posse de telefone móvel celular para uso pessoal 2025**. Rio de Janeiro, 2026. Disponível em: https://www.ibge.gov.br/estatisticas/sociais/populacao/17270-pnad-continua.html. Acesso em: 5 ago. 2026. Citado na p. 10.

ISINKAYE, Folasade Olubusola. Matrix Factorization in Recommender Systems: Algorithms, Applications, and Peculiar Challenges. **IETE Journal of Research**, 2021. DOI: 10.1080/03772063.2021.1997357. Citado na p. 16.

JONES, Karen Spärck. A Statistical Interpretation of Term Specificity and Its Application in Retrieval. **Journal of Documentation**, v. 28, n. 1, p. 11–21, 1972. DOI: 10.1108/eb026526. Citado nas pp. 15, 26, 34.

KO, Hyeyoung *et al.* A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields. **Electronics**, v. 11, n. 1, p. 141, 2022. DOI: 10.3390/electronics11010141. Citado nas pp. 14, 16.

KOREN, Yehuda; BELL, Robert; VOLINSKY, Chris. Matrix Factorization Techniques for Recommender Systems. **Computer**, v. 42, n. 8, p. 30–37, 2009. DOI: 10.1109/MC.2009.263. Citado nas pp. 16, 47.

LOPS, Pasquale; GEMMIS, Marco de; SEMERARO, Giovanni. Content-based Recommender Systems: State of the Art and Trends. *In:* RICCI, Francesco *et al.* (ed.). **Recommender Systems Handbook**. Boston, MA: Springer, 2011. p. 73–105. DOI: 10.1007/978-0-387-85820-3_3. Citado nas pp. 14, 25, 33.

MANNING, Christopher D.; RAGHAVAN, Prabhakar; SCHÜTZE, Hinrich. **Introduction to Information Retrieval**. Cambridge: Cambridge University Press, 2008. Citado nas pp. 15, 27, 35, 47.

MEZA, Laura Romero; D'URSO, Giulio. User's Dilemma: A Qualitative Study on the Influence of Netflix Recommender Systems on Choice Overload. **Psychological Studies**, v. 69, n. 3, p. 349–367, 2024. DOI: 10.1007/s12646-024-00807-0. Citado nas pp. 10, 13, 46.

NIELSEN. **2023 state of play report: data-driven personalization: the future of streaming content discovery**. New York, 2023. Disponível em: https://www.nielsen.com/insights/2023/data-driven-personalization-2023-state-of-play-report/. Acesso em: 5 ago. 2026. Citado na p. 10.

PANDA, Deepak Kumar; RAY, Sanjog. Approaches and Algorithms to Mitigate Cold Start Problems in Recommender Systems: A Systematic Literature Review. **Journal of Intelligent Information Systems**, v. 59, n. 2, p. 341–366, 2022. DOI: 10.1007/s10844-022-00698-5. Citado na p. 16.

PAPADAKIS, Harris *et al.* Content-Based Recommender Systems Taxonomy. **Foundations of Computing and Decision Sciences**, v. 48, n. 2, p. 211–241, 2023. DOI: 10.2478/fcds-2023-0009. Citado na p. 14.

PAZZANI, Michael J.; BILLSUS, Daniel. Content-Based Recommendation Systems. *In:* THE Adaptive Web. Berlin: Springer, 2007. p. 325–341. DOI: 10.1007/978-3-540-72079-9_10. Citado na p. 14.

PERMANA, Armadhani Hiro Juni; WIBOWO, Agung Toto. Movie Recommendation System Based on Synopsis Using Content-Based Filtering with TF-IDF and Cosine

Similarity. **International Journal on Information and Communication Technology (IJoICT)**, v. 9, n. 2, p. 1–14, 2023. DOI: 10.21108/ijoict.v9i2.747. Citado na p. 15.

REIMERS, Nils; GUREVYCH, Iryna. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *In:* PROCEEDINGS of the 2019 Conference on Empirical Methods in Natural Language Processing. [*S. l.*]: Association for Computational Linguistics, 2019. p. 3982–3992. DOI: 10.18653/v1/D19-1410. Citado nas pp. 15, 16, 28, 47.

RESNICK, Paul; VARIAN, Hal R. Recommender Systems. **Communications of the ACM**, v. 40, n. 3, p. 56–58, 1997. DOI: 10.1145/245108.245121. Citado nas pp. 13, 46.

RICCI, Francesco; ROKACH, Lior; SHAPIRA, Bracha (ed.). **Recommender Systems Handbook**. 2. ed. Boston: Springer, 2015. DOI: 10.1007/978-1-4899-7637-6. Citado nas pp. 14, 46.

SALTON, Gerard; MCGILL, Michael J. **Introduction to Modern Information Retrieval**. New York: McGraw-Hill, 1983. Citado nas pp. 15, 26, 34.

SCHEIN, Andrew I. *et al.* Methods and Metrics for Cold-Start Recommendations. *In:* PROCEEDINGS of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. [*S. l.*]: ACM, 2002. p. 253–260. DOI: 10.1145/564376.564421. Citado nas pp. 16, 19, 42, 47.

SCHWARTZ, Barry. **The Paradox of Choice: Why More Is Less**. New York: Harper Perennial, 2004. Citado nas pp. 10, 13, 46.

Emitido em 11/08/2026

DOCUMENTO Nº 1/2026 - CCDN (11.00.52.09)
(Nº do Documento: 1)

(Nº do Protocolo: NÃO PROTOCOLADO)

(Assinado digitalmente em 26/08/2026 10:29)
ANDREA ALVES BORBA
ASSISTENTE EM ADMINISTRACAO
1517808

Para verificar a autenticidade deste documento entre em https://sipac.ufpb.br/documentos/ informando seu número: **1**, ano: **2026**, documento (espécie): **DOCUMENTO**, data de emissão: **26/08/2026** e o código de verificação: **9f92397f6b**