FACULDADE DE TECNOLOGIA DE BAURU

TECNOLOGIA EM BANCO DE DADOS

SISTEMA DE RECOMENDAÇÃO DE FILMES BASEADO EM CIÊNCIA DE DADOS: Uma Análise Técnica E Reprodução De Soluções Aplicadas

Breno Lucas de Oliveira
João Gabriel Manhoni
Victor Beltrão Baptistella

Bauru/SP
2025

SISTEMA DE RECOMENDAÇÃO DE FILMES BASEADO EM CIÊNCIA DE DADOS: Uma Análise Técnica E Reprodução De Soluções Aplicadas

Breno Lucas de Oliveira
João Gabriel Manhoni
Victor Beltrão Baptistella

Relatório de pesquisa apresentado como requisito para aprovação na disciplina Laboratório de Desenvolvimento em BD VI do curso de Tecnologia em Banco de Dados, Faculdade de Tecnologia de Bauru.

Profa. Dra. Patricia Bellin Ribeiro

Bauru/SP
2025

SUMÁRIO

RESUMO

Os sistemas de recomendação são componentes fundamentais na indústria moderna de entretenimento digital, pois personalizam a experiência do usuário e impulsionam a descoberta de novos conteúdos. Este estudo tem como foco a análise técnica e a reprodução de um sistema de recomendação de filmes baseado em conteúdo. Utilizando o conjunto de dados TMDb 5000 movies, o projeto foi desenvolvido em Python, com o auxílio das bibliotecas Pandas e Scikit-learn. A metodologia central envolveu o processamento das sinopses dos filmes (overview) por meio da técnica Term Frequency–Inverse Document Frequency (TF-IDF), a fim de converter os dados textuais em uma representação vetorial numérica significativa. Em seguida, foi aplicado o kernel sigmoide para calcular uma matriz de similaridade entre todos os filmes. Os resultados demonstram a eficácia do sistema em gerar recomendações relevantes e coerentes, validadas por meio de testes com filmes como Avatar, The Matrix e Spectre. Este trabalho contribui ao detalhar uma implementação prática de um modelo de filtragem baseado em conteúdo, discutindo seus resultados, limitações e potencial para aprimoramentos futuros.

ABSTRACT

Recommendation systems are fundamental components in the modern digital entertainment industry, personalizing user experience and driving content discovery. This study focuses on the technical analysis and reproduction of a content-based movie recommendation system. Utilizing the TMDb 5000 movie dataset, the project was developed in Python with the aid of the Pandas and Scikit-learn libraries. The core methodology involved processing movie synopses (overview) using the Term Frequency-Inverse Document Frequency (TF-IDF) technique to convert textual data into a meaningful numerical vector representation. Subsequently, the sigmoid kernel was applied to compute a similarity matrix between all films. The results demonstrate the system's effectiveness in generating relevant and coherent recommendations, validated through test cases with films such as 'Avatar', 'The Matrix', and 'Spectre'. This work contributes by detailing a practical implementation of a content-based filtering model, discussing its results, limitations, and potential for future enhancements.

1. INTRODUÇÃO

Os sistemas de recomendação representam uma das aplicações mais populares e impactantes da ciência de dados na era digital contemporânea (AGGARWAL, 2016). Estes sistemas são projetados para predizer a avaliação ou preferência que um usuário atribuiria a um determinado item, utilizando algoritmos sofisticados de aprendizado de máquina e análise de dados para personalizar a experiência do usuário (JANNACH et al., 2010).

A relevância dos sistemas de recomendação no cenário tecnológico atual é evidenciada pela sua ampla adoção por grandes corporações (RICCI; ROKACH; SHAPIRA, 2011). A Amazon utiliza estes sistemas para sugerir produtos aos seus clientes, aumentando significativamente as taxas de conversão e satisfação do consumidor. O YouTube implementa algoritmos de recomendação para determinar qual vídeo será reproduzido automaticamente na sequência, mantendo os usuários engajados na plataforma por períodos mais prolongados. O Facebook emprega tecnologias similares para recomendar

páginas para curtir e pessoas para seguir, fortalecendo as conexões sociais dentro da rede.

No contexto da indústria cinematográfica e de entretenimento, os sistemas de recomendação de filmes desempenham um papel crucial na descoberta de conteúdo. Plataformas como Netflix, Amazon Prime Video e Disney+ investem milhões de dólares no desenvolvimento e aprimoramento destes sistemas, reconhecendo que a capacidade de sugerir conteúdo relevante e personalizado é fundamental para a retenção de usuários e o sucesso comercial.

O presente trabalho foca na análise e reprodução de um sistema de recomendação de filmes baseado em conteúdo, utilizando técnicas de processamento de linguagem natural e aprendizado de máquina. Este sistema analisa as sinopses dos filmes (campo "overview") para identificar similaridades temáticas e narrativas entre diferentes produções cinematográficas.

A abordagem de recomendação baseada em conteúdo (Content-Based Filtering) representa uma das metodologias fundamentais na área, diferenciando-se de outras técnicas como filtragem colaborativa por não depender do comportamento de outros usuários (FELFERNIG et al., 2011). Em vez disso, este método analisa as características intrínsecas dos itens - neste caso, os aspectos narrativos e temáticos dos filmes - para estabelecer relações de similaridade.

O *dataset* utilizado neste projeto provém do The Movie Database (TMDb), uma base de dados cinematográfica colaborativa que contém informações detalhadas sobre milhares de filmes (THE MOVIE DATABASE, 2025). Os dados incluem não apenas informações básicas como título e ano de lançamento, mas também metadados ricos como sinopses, informações de elenco, orçamento e receita, proporcionando um ambiente rico para experimentação e análise.

Do ponto de vista técnico, o sistema implementa a técnica TF-IDF (Term Frequency-Inverse Document Frequency) para transformar as sinopses textuais em representações numéricas que podem ser processadas por algoritmos de aprendizado de máquina (SALTON; BUCKLEY, 1988; MANNING; RAGHAVAN; SCHÜTZE, 2008). Esta abordagem permite quantificar a importância de palavras específicas dentro do contexto das descrições dos filmes, identificando termos que são distintivos para cada produção.

Posteriormente, utiliza-se o *kernel* sigmoide para calcular medidas de similaridade entre os filmes, criando uma matriz de similaridade que serve como base para as recomendações. A implementação técnica é realizada utilizando a biblioteca scikit-learn (PEDREGOSA *et al*., 2011; SCIKIT-LEARN DEVELOPERS, 2025), que fornece ferramentas robustas para processamento de texto e cálculo de similaridades. Esta escolha metodológica permite capturar relações não-lineares entre as características dos filmes, potencialmente resultando em recomendações mais precisas e nuançadas.

Este estudo baseia-se no projeto desenvolvido por Aiman (2020), que apresenta uma implementação prática e didática de um sistema de recomendação de filmes. O trabalho não apenas implementa e analisa o sistema de recomendação proposto, mas também realiza uma engenharia reversa do código, examinando cada componente algorítmico, suas justificativas teóricas e implicações práticas. O objetivo é proporcionar uma compreensão abrangente dos fundamentos técnicos e teóricos que sustentam os sistemas de recomendação modernos.

2. OBJETIVOS

2.1 Objetivo Geral

O presente trabalho tem como objetivo principal analisar, compreender e reproduzir um sistema de recomendação de filmes baseado em conteúdo, implementado em Python (AIMAN, 2020), realizando uma análise técnica detalhada dos algoritmos utilizados e dos resultados obtidos, com foco na aplicação prática de técnicas de ciência de dados e aprendizado de máquina. Esta investigação busca proporcionar uma compreensão aprofundada dos fundamentos teóricos e práticos que sustentam os sistemas de recomendação modernos, contribuindo para o desenvolvimento de competências técnicas na área de ciência de dados aplicada ao entretenimento digital.

2.2 Objetivos Específicos

Para alcançar o objetivo geral proposto, este estudo se desdobra em múltiplas dimensões de análise que abrangem desde a compreensão conceitual até a implementação prática do sistema. Inicialmente, busca-se realizar uma leitura crítica e análise técnica minuciosa do código-fonte do sistema de recomendação proposto, identificando e documentando as bibliotecas, algoritmos e metodologias empregadas na implementação. Esta etapa fundamental envolve a compreensão da arquitetura geral do sistema e do fluxo de processamento de dados, estabelecendo as bases para as análises subsequentes.

A dimensão experimental do trabalho concentra-se na execução do código original seguindo rigorosamente a metodologia de engenharia reversa, reproduzindo todos os experimentos apresentados no projeto base e validando os resultados obtidos através de comparações sistemáticas com os resultados esperados. Esta abordagem metodológica permite não apenas verificar a funcionalidade do sistema, mas também compreender profundamente cada etapa do processo de desenvolvimento.

No aspecto técnico-algorítmico, o estudo se propõe a examinar detalhadamente a aplicação da técnica TF-IDF para processamento de texto (SALTON; BUCKLEY, 1988), analisando sua implementação específica no contexto de recomendação de filmes. Paralelamente, investigar-se-á a utilização do kernel sigmoide para cálculo de similaridade entre filmes, avaliando a eficácia da função de recomendação implementada utilizando as ferramentas disponibilizadas pela biblioteca scikit-learn (PEDREGOSA et al., 2011). Esta análise técnica visa compreender as escolhas algorítmicas e suas implicações na qualidade das recomendações geradas.

O trabalho também dedica atenção especial ao processamento e preparação de dados, analisando criteriosamente as etapas de limpeza e preparação do dataset TMDb, compreendendo o processo de fusão entre os datasets de filmes e créditos, e avaliando como as decisões de pré-processamento influenciam diretamente os resultados finais do sistema. Esta dimensão é crucial para compreender a importância da qualidade dos dados em sistemas de aprendizado de máquina.

A avaliação de resultados constitui outro pilar fundamental do estudo, envolvendo testes sistemáticos do sistema com diferentes filmes de entrada, avaliação crítica da qualidade e relevância das recomendações geradas, e identificação de possíveis limitações e áreas de melhoria do sistema proposto. Esta análise crítica contribui para uma compreensão realista das capacidades e limitações da abordagem implementada.

Do ponto de vista acadêmico e científico, o trabalho visa documentar meticulosamente todo o processo de análise e reprodução do sistema, elaborando um relatório técnico detalhado sobre os experimentos realizados e apresentando conclusões fundamentadas sobre a eficácia e aplicabilidade do sistema estudado. Esta

documentação serve não apenas como registro do trabalho realizado, mas também como contribuição para futuras investigações na área.

Finalmente, busca-se estabelecer uma sólida fundamentação teórica contextualizando o projeto dentro do campo mais amplo dos sistemas de recomendação (Aggarwal, 2016; Jannach et al., 2010), comparando a abordagem baseada em conteúdo com outras metodologias existentes (Felfernig et al., 2011), e discutindo as implicações práticas e comerciais dos sistemas de recomendação na indústria contemporânea (Ricci; Rokach; Shapira, 2011). Esta contextualização teórica é essencial para compreender a relevância e o posicionamento do trabalho dentro do panorama atual da ciência de dados aplicada.

3. MATERIAIS E MÉTODOS

3.1 Ambiente de Desenvolvimento

O desenvolvimento do projeto foi conduzido na linguagem de programação Python (versão 3.12), reconhecida por sua robustez e vasto ecossistema de bibliotecas para ciência de dados. O ambiente de execução utilizado foi o Google Colaboratory (Colab), uma plataforma baseada em nuvem que fornece acesso a recursos computacionais e um ambiente de notebooks interativos, facilitando a execução e a reprodutibilidade do código. As principais bibliotecas utilizadas foram:

- **Pandas:** Para a manipulação e análise de dados, sendo fundamental na leitura dos arquivos, fusão dos *datasets* e estruturação das informações em *DataFrames*.
- **Scikit-learn:** A biblioteca central para a implementação dos algoritmos de aprendizado de máquina. Foram utilizados especificamente o módulo TfidfVectorizer para a extração de características textuais e a função sigmoid_kernel para o cálculo da similaridade.

3.2 Coleta e Descrição dos Dados

O estudo utilizou o conjunto de dados público "TMDb 5000 Movie Dataset", obtido na plataforma de competições de ciência de dados Kaggle. Este dataset é composto por dois arquivos em formato CSV:

1. **tmdb_5000_movies.csv:** Contém informações detalhadas de aproximadamente 5.000 filmes, incluindo colunas como id (identificador único), title (título) e overview (sinopse).
2. **tmdb_5000_credits.csv:** Contém os créditos associados a cada filme, incluindo elenco (*cast*) e equipe técnica (*crew*), vinculados pelo identificador do filme.

A característica fundamental para este sistema de recomendação baseado em conteúdo é a coluna overview, que fornece a descrição textual da narrativa e temática de cada filme.

3.3 Pré-processamento e Preparação dos Dados

Antes da aplicação dos algoritmos, os dados brutos foram submetidos a um processo de limpeza e preparação para garantir a consistência e a qualidade das informações utilizadas na modelagem. As etapas foram as seguintes:

1. **Carregamento e Fusão:** Os dois arquivos CSV foram carregados como DataFrames da biblioteca Pandas. Em seguida, foram fundidos em um único DataFrame utilizando o identificador do filme como chave de junção.

2. **Seleção de Características (Feature Selection):** Para a abordagem de recomendação baseada em conteúdo, a característica principal selecionada foi a sinopse (overview). As demais colunas foram mantidas para identificação e apresentação dos resultados.
3. **Tratamento de Dados Ausentes:** Foi verificado que algumas entradas na coluna *overview* possuíam valores nulos (NaN). Para evitar erros durante a etapa de vetorização, esses valores foram substituídos por uma *string* vazia ('').

3.4 Modelagem e Implementação do Sistema de Recomendação

O núcleo do sistema de recomendação foi implementado seguindo uma abordagem de filtragem baseada em conteúdo, que analisa as características intrínsecas dos itens para sugerir filmes similares.

1. **Vetorização de Texto com TF-IDF:** Para que os algoritmos pudessem processar as sinopses, foi necessário converter o conteúdo textual em uma representação numérica. Para isso, foi utilizada a técnica TF-IDF (*Term Frequency-Inverse Document Frequency*), implementada pela classe TfidfVectorizer do Scikit-learn. Este método calcula um peso para cada palavra em cada sinopse, atribuindo maior importância aos termos que são frequentes em um documento, mas raros no conjunto de todos os documentos (corpus). O processo resultou em uma matriz onde cada linha representava um filme e cada coluna um termo ponderado pelo seu score TF-IDF.
2. **Cálculo de Similaridade com Kernel Sigmoide:** Com os filmes representados como vetores TF-IDF, o passo seguinte foi calcular a similaridade entre cada par de filmes. Para esta tarefa, foi aplicada a função Kernel Sigmoide (sigmoid_kernel do Scikit-learn) sobre a matriz TF-IDF. O resultado foi uma matriz de similaridade quadrada, na qual o elemento na posição (i, j) representa o *score* de similaridade entre o filme i e o filme j.
3. **Geração das Recomendações:** A lógica final para gerar as recomendações foi implementada da seguinte forma:
    - O sistema recebe o título de um filme como entrada.
    - Localiza o índice deste filme na matriz de similaridade.
    - Enumera e ordena os scores de similaridade de todos os outros filmes em relação ao filme de entrada, em ordem decrescente.
    - Os 10 filmes com os maiores scores de similaridade são selecionados e retornados como a recomendação final, excluindo-se o próprio filme de entrada.

4. RESULTADOS E DISCUSSÃO

Esta seção apresenta os resultados obtidos a partir da execução da metodologia descrita, seguida de uma discussão aprofundada sobre as implicações e a eficácia do sistema de recomendação desenvolvido.

4.1. Processamento e Análise dos Dados

A execução das etapas de pré-processamento foi bem-sucedida. Inicialmente, os datasets credits e movies foram carregados, apresentando as dimensões de (4803, 4) e (4803, 20), respectivamente. Após a fusão dos dois dataframes com base na coluna 'id', um único dataset consolidado foi criado.

A etapa de limpeza envolveu a remoção de colunas desnecessárias para o modelo de conteúdo, como homepage, title_x, title_y, status e production_countries. Um passo cru-

cial foi o tratamento de dados ausentes na coluna overview, onde valores nulos (NaN) foram substituídos por strings vazias ('). Esta ação foi fundamental para prevenir o ValueError durante a vetorização e garantir que todos os 4803 registros pudessem ser processados pelo modelo.

```
      movies_cleaned['overview'] = movies_cleaned['overview'].fill-
na('')
      tfv_matrix = tfv.fit_transform(movies_cleaned['overview'])
      print(tfv_matrix)
      print(tfv_matrix.shape)
```

Tabela 1: Descrição das *Features* Utilizadas no Modelo Final

Feature	Descrição	Exemplo de uso
original_title	Título original do filme.	Utilizado como entrada para o usuário e na exibição dos resultados
overview	Sinopse textual do filme	Principal característica para a análise de conteúdo e cálculo de similaridade
id	Identificador único do filme	Chave primária para a fusão dos *datasets* iniciais

4.2. Vetorização e Cálculo de Similaridade

A aplicação do TfidfVectorizer na coluna overview resultou na criação de uma matriz esparsa de (4803, 10417). Esta dimensão indica que o sistema analisou 4803 filmes e identificou um vocabulário de 10.417 termos únicos e relevantes (considerando os n-gramas de 1 a 3 palavras) para representá-los numericamente.

Posteriormente, o sigmoid_kernel foi aplicado a essa matriz, gerando uma matriz de similaridade densa de dimensões (4803, 4803). Cada célula (i, j) desta matriz contém um score que quantifica o quão similar o filme i é do filme j, com base exclusivamente no conteúdo de suas sinopses.

4.3. Validação das Recomendações Geradas

O teste final do sistema consistiu em fornecer títulos de filmes conhecidos como entrada para a função give_recomendations. Os resultados para três filmes distintos são apresentados e analisados a seguir.

```
      print(give_recomendations('Avatar'))
```

Tabela 2: Recomendações Geradas para o Filme "Avatar"

Ranking	Título do Filme Recomendado
1	Obitaemyy Ostrov
2	The Matrix
3	Apollo 18
4	The American
5	Supernova
6	Tears of the Su

7	Beowulf
8	The Adventures of Pluto Nash
9	Semi-Pro
10	The Book of Life

As recomendações para "Avatar" apontam para filmes de ficção científica ("Obitaemyy Ostrov", "Apollo 18", "Supernova") e fantasia/ação ("The Matrix", "Beowulf"). Isso demonstra que o modelo conseguiu extrair com sucesso o núcleo temático da sinopse de "Avatar", que envolve mundos alienígenas e conflitos, e encontrar outros filmes com um vocabulário similar.

```
print(give_recomendations('The Matrix'))
```

Tabela 3: Recomendações Geradas para o Filme "The Matrix"

Ranking	**Título do Filme Recomendado**
1	Pulse
2	Avatar
3	Obitaemyy Ostrov
4	Supernova
5	The Specials
6	Hackers
7	Commando
8	The Girl with the Dragon Tattoo
9	Oliver
10	The Curious Case of Benjamin Button

O sistema recomendou filmes que compartilham temas de tecnologia, realidade simulada e ação, como "Hackers" e "Pulse". A inclusão de "Avatar" e "Obitaemyy Ostrov" reforça a conexão com o gênero de ficção científica. A presença de filmes de ação como "Commando" sugere que o vocabulário relacionado a conflitos e combate também foi um fator de peso na similaridade.

```
print(give_recomendations('Spectre'))
```

Tabela 4: Recomendações Geradas para o Filme "Spectre"

Ranking	**Título do Filme Recomendado**
1	Never Say Never Again
2	Skyfall
3	Thunderball
4	From Russia with Love
5	Quantum of Solace
6	The Man with the Golden Gun
7	Safe Haven
8	2016: Obama's America
9	The Living Daylights
10	Dr. No

Este é o resultado mais forte e uma validação clara da eficácia do modelo. Das 10 recomendações, 8 são outros filmes da franquia James Bond. Isso ocorre porque as si-

nopses dos filmes de Bond compartilham um vocabulário extremamente específico (nomes de personagens como "Bond", agências como "MI6", termos como "agente", "missão", "espião"), que o TF-IDF identifica corretamente como sendo de alta relevância, resultando em altos scores de similaridade entre eles.

5. CONCLUSÃO

O presente trabalho teve como objetivo analisar, reproduzir e avaliar um sistema de recomendação de filmes baseado em conteúdo. Através da aplicação de técnicas de Processamento de Linguagem Natural (TF-IDF) e aprendizado de máquina (Kernel Sigmoide) sobre um *dataset* público do TMDb, foi possível implementar com sucesso um modelo funcional.

Os objetivos específicos foram plenamente alcançados. A engenharia reversa do código permitiu uma compreensão aprofundada da arquitetura do sistema e do fluxo de dados. A execução dos experimentos validou a metodologia, demonstrando que a análise de sinopses é uma abordagem eficaz para capturar a similaridade temática entre filmes. Os resultados, especialmente no caso do filme "Spectre", onde o sistema recomendou majoritariamente outros títulos da mesma franquia, servem como uma forte evidência da precisão do modelo em agrupar conteúdos contextualmente relacionados.

Contudo, o estudo também evidencia as limitações de uma abordagem puramente baseada em conteúdo textual. O sistema não possui conhecimento sobre diretores, elenco, gênero (a menos que citados na sinopse) ou a opinião de outros usuários, que são fatores importantes na decisão de um espectador.

Como trabalhos futuros, sugere-se a expansão do modelo para um sistema híbrido. Isso poderia envolver a incorporação de outras features, como gênero, elenco e diretor, no processo de cálculo de similaridade, ou a combinação da abordagem de conteúdo com técnicas de filtragem colaborativa, que analisam o comportamento e as avaliações de outros usuários para gerar recomendações mais personalizadas e diversificadas.

6. REFERÊNCIAS

AIMAN. Data Scie ce Project — Movie Recommendation System. Aiman's AI, 20 maio 2020. Disponível em: https://amanxai.com/2020/05/20/data-science-project-movie-recommendation-system/. Acesso em: 22 set. 2025.

AGGARWAL, C. C. Recommender Systems: The Textbook. New York: Springer, 2016.

FELFERNIG, A. et al. An Introduction to Recommender Systems. Cambridge: Cambridge University Press, 2011.

JANNACH, D. et al. Recommender Systems: An Introduction. Cambridge: Cambridge University Press, 2010.

MANNING, C. D.; RAGHAVAN, P.; SCHÜTZE, H. Introduction to Information Retrieval. Cambridge: Cambridge University Press, 2008.

PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, v. 12, p. 2825-2830, 2011.

RICCI, F.; ROKACH, L.; SHAPIRA, B. Introduction to Recommender Systems Handbook. Boston: Springer, 2011.

SALTON, G.; BUCKLEY, C. Term-weighting approaches in automatic text retrieval. Information Processing & Management, v. 24, n. 5, p. 513-523, 1988.

SCIKIT-LEARN DEVELOPERS. TF-IDF Vectorizer Documentation. Scikit-learn. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html. Acesso em: 22 set. 2025.

THE MOVIE DATABASE (TMDb). TMDb 5000 Movie Dataset. Kaggle. Disponível em: https://www.kaggle.com/tmdb/tmdb-movie-metadata. Acesso em: 22 set. 2025.