

Inteligência Artificial - Sistema de Recomendação de Filmes (RBC)



Estudantes

Johnny Matheus Nogueira de Medeiro

Nathaniel Nicolas Rissi Soares

Nelson Ramos Rodrigues Junior



Instituição

Ciências da Computação

UNOESC - São Miguel do Oeste

Visão Geral do Projeto

O projeto desenvolvido consiste em um sistema de recomendação de filmes baseado em Raciocínio Baseado em Casos (RBC), criado utilizando modernas ferramentas de desenvolvimento e conceitos de inteligência artificial, aplicando todo o conhecimento adquirido no componente curricular Inteligência Artificial do curso de Ciência da Computação da Universidade do Oeste de Santa Catarina (UNOESC), ministrado pelo professor Vinicius Almeida dos Santos.

O objetivo principal do projeto é aplicar na prática os conhecimentos adquiridos em sala de aula, fornecendo recomendações personalizadas de filmes com base em avaliações, votos e ano de lançamento. O sistema foi implementado utilizando Python, Pandas, Numpy, Scikit-learn, Streamlit e ferramentas de visualização interativa, proporcionando uma experiência prática e didática na aplicação de técnicas de inteligência artificial em dados reais.



Raciocínio Baseado em Casos

O RBC funciona como o raciocínio humano: você compara um novo caso (filme que o usuário gosta) com casos anteriores (outros filmes da base) e recomenda os mais similares.



Objetivo Principal

Ajudar o usuário a descobrir filmes parecidos com os que ele já gosta, usando informações numéricas e características dos filmes para gerar recomendações personalizadas.



Base de Dados

A base utilizada contém informações completas sobre filmes:

- **Title:** Nome do filme
- **IMDb Rating:** Nota do público
- **Meta Score:** Nota da crítica
- **Votes:** Número de votos
- **Year:** Ano de lançamento
- **Genre:** Gêneros do filme
- **Tags:** Palavras-chave/temas
- **Director:** Diretor
- **Stars:** Elenco principal
- **Poster URL:** Imagem do filme

▲ Normalização Min-Max

Foi utilizada a normalização Min-Max (também chamada de escala 0–1), implementada pelo MinMaxScaler da biblioteca Scikit-learn. A ideia é ajustar os valores de cada feature (atributo numérico) para que todos fiquem no mesmo intervalo, entre 0 e 1.

Problema sem Normalização

Atributos com valores muito grandes (como votos) iriam dominar o cálculo da similaridade, prejudicando a precisão das recomendações.

Solução com Min-Max

Todos os valores ficam comparáveis dentro do intervalo [0,1], garantindo que cada característica tenha peso equilibrado no algoritmo.

Exemplo Prático:

- [IMDb Rating](#) → varia de 1 a 10
- [Meta Score](#) → varia de 0 a 100
- [Votes](#) → pode variar de centenas até milhões
- [Ano de lançamento](#) → varia de 1920 até 2020

◆ Fórmula utilizada

A fórmula da normalização Min-Max é:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Onde:

- x → valor original
- x_{min} → menor valor da coluna
- x_{max} → maior valor da coluna
- x' → valor normalizado (sempre entre 0 e 1)

Implementação e Funcionalidades

Exemplo prático

Se o IMDb Rating vai de 1 a 10:

$$x' = \frac{x - 1}{10 - 1}$$

- Filme com rating 7.5 →

$$x' = \frac{7.5 - 1}{9} = 0.72$$

- Filme com rating 9.0 →

$$x' = \frac{9.0 - 1}{9} = 0.89$$

Assim, todos os atributos ficam na mesma escala antes de calcular a similaridade de cosseno.

Pré-processamento dos Dados

01

Limpeza de Votos

Transforma valores com "K" em números inteiros para padronização dos dados.

02

Extração do Ano

Pega o ano do título do filme para análise temporal das preferências.

03

Seleção de Features

IMDb Rating, Meta Score, votos e ano são selecionados como características numéricas.

04

Normalização

Usa MinMaxScaler para colocar todas as características na mesma escala (0 a 1).

Algoritmo de Recomendação

O coração do projeto é a função de recomendação que utiliza similaridade de cosseno para medir o quão parecidos dois filmes são considerando seus valores numéricos.

1

Seleção do Filme

O usuário seleciona um filme na interface interativa

2

Busca na Base

O sistema encontra o filme na base de dados (busca parcial e case insensitive)

3

Cálculo de Similaridade

Calcula a similaridade de cosseno entre o filme escolhido e todos os outros

4

Recomendações

Retorna os top K filmes mais similares com nota e gênero

Interface Streamlit

Funcionalidades Interativas

- Selectbox:** permite escolher um filme
- Slider:** define quantas recomendações mostrar
- Checkboxes:** mostrar dados brutos ou debug
- Sidebar:** estatísticas da base e exploração por gênero

Tratamento de Erros

- Se o CSV não existir → mostra mensagem de erro
- Se uma coluna faltar → usa apenas as colunas disponíveis
- Se o filme não for encontrado → avisa o usuário

🧠 Equipe de Desenvolvimento



👑 Johnny Matheus

[Game Master - Johnny Matheus Nogueira de Medeiro](#)



Nelson Ramos

[Nelson Ramos Rodrigues Junior](#)



Nathaniel Nicolas

[Nathaniel Nicolas Rissi Soares](#)



Professor Orientador

[Vinicius Almeida dos Santos](#) - Responsável pelo componente curricular de Inteligência Artificial na UNOESC