

# Sistema de Recomendação de Filmes

Adriana Lopes, Diana Carrilho, Henrique Faria, Paulo Barbosa

Universidade do Minho, Mestrado Integrado em Engenharia Informática e Mestrado em Matemática e Computação

**Resumo** Neste trabalho abordou-se o tema "Sistemas de Recomendação". Este projeto insidiu sobre um dataSet de filmes encontrado no Kaggle e guardado em ficheiros excel. Para a realização deste projeto usaram-se várias linguagens de programação e tecnologias, nomeadamente Flask(Python), HTML, CSS, Jinja2, JavaScript e MongoDB. O Mongo foi usado para guardar o nosso dataSet de forma a que este no futuro possa escalar horizontalmente e também possibilitar um acesso mais rápido aos dados. O HTML, o CSS, o Jinja2 e o JavaScript foram usados para fazer as páginas web e programar o seu comportamento. O Flask foi usado para fazer a ligação entre FrontEnd e Backend e para controlar as mudanças de estado das páginas web.

**Palavras-chave:** Sistemas de Recomendação · Flask · Python · HTML · CSS · Jinja2 · JavaScript · MongoDB · Filmes

## Introdução

Neste trabalho foi explorada a aplicação dos Sistemas de Recomendação a uma base de dados de filmes. O primeiro passo para este objetivo passou por planear os requisitos dos utilizadores da nossa base de dados que usem o nosso sistema de recomendação. Estes requisitos são: obter uma recomendação baseada no que os top X utilizadores semelhantes ao alvo viram e obter uma recomendação baseada no conteúdo que o alvo viu e gostou (por exemplo por género de filme). Para os casos de cold-start de utilizadores foram adicionados métodos para apresentação dos top 10 filmes mais bem cotados pela IMDB e os 15 filmes mais bem cotados por utilizadores do sistema e para os casos de cold-start dos filmes foi criado um método para devolver os 15 filmes mais recente ainda sem visualizações por data de lançamento(do mais recente para o mais antigo). Para além disso foi introduzido uma search bar que permite ao utilizador da aplicação utilizar um sistema de recomendação baseado em restrições, iterativo. Adicionalmente juntaram-se os requisitos de consultar filmes por ano de lançamento ou por género.

Para permitir uma experiência personalizada aos utilizadores foi necessário identificá-los inequivocamente e para isso tivemos de implementar um sistema de autenticação no qual o utilizador é identificado por um email único e ao qual é associado um id usado para guardar as suas avaliações dos filmes vistos.

## 1 Objetivos

Calcular `speed_diff`

## 2 Conclusão

Na aplicação apresentada são usadas várias abordagens aos sistemas de recomendação com o intuito de fornecer as melhores recomendações de filmes nomeadamente filtros colaborativos baseados em utilizadores, recomendações baseadas em conteúdo assistido e recomendações baseadas em restrições como palavras ou frases. Para além disso, ainda se implementaram 3 recomendações não pertencentes às estudadas para melhorar a experiência ao utilizar a aplicação e para resolver o problema de cold-case de filmes (os 10 filmes mais bem cotados pelo IMDB, filmes mais bem cotados pelos utilizadores e filmes recentes ainda sem visualizações). Ainda que o trabalho cumpra o requisito de aplicar os sistemas de recomendação aos filmes e esteja pensado para escalar horizontalmente, não trata aspetos como a segurança. Implementar mecanismos como recuperação de conta e encriptação de mensagens trocadas entre cliente e servidor ficarão para trabalho futuro. Em suma, a aplicação desenvolvida responde às necessidades de recomendação de filmes por parte de qualquer utilizador.