

# Sistema de Recomendação de Filmes

Patrícia Cordeiro Pereira Pampanelli

*São Paulo, Brasil*

---

## Abstract

*Keywords:* recommendation, movies, data science, content-based recommender

---

## 1. Introdução

Sistemas de recomendação são amplamente conhecidos e utilizados em diversas aplicações. Estes sistemas tem como objetivo principal indicar produtos/serviços/conteúdos com base em perfis ou características de produtos já conhecidos. Neste sentido, os sistemas de recomendação podem ser divididos em dois tipos: - filtragem colaborativa; - filtragem baseada em conteúdo.

A filtragem colaborativa consiste na utilização de informações provenientes de comportamentos, atividades e preferências e, consequentemente, a análise de similaridade com outros usuários. Tendo como base esta análise comportamental são indicados novos produtos e serviços. A base para estas recomendações é que usuários que têm gostos similares tem grande probabilidade de concordar em recomendações futuras. Em outras palavras, os sistemas de recomendação baseados em filtragem colaborativa são construídos com base no comportamento dos usuários e suas similaridades. Um exemplo clássico deste tipo de recomendação é a sugestão de filmes tendo como base o perfil de cada usuário, os filmes já assistidos e os ratings fornecidos pelos mesmos.

Os sistemas de recomendação baseados em conteúdo buscam conhecer os produtos e serviços que serão objeto de recomendações. Por exemplo, para recomendação de filmes com base no conteúdo são analisadas as categorias de cada filme, o estilo, os atores e diretores e etc. Neste tipo de recomendação o foco é o conteúdo e suas características. Estes sistemas tem como premissa que o usuário irá gostar de novos produtos e serviços similares ao filmes que receberam altos ratings anteriormente.

25 Além disso, existem os sistemas híbridos que combinam características  
26 das recomendações colaborativas e dos sistemas baseados em conteúdo para  
27 obter melhores resultados.

## 28 **2. Definição do Problema**

29 O problema que será abordado neste projeto é a recomendação de filmes,  
30 ou seja, tendo como base um conjunto de características de usuários e/ou  
31 filmes recomendar novos itens que ainda não foram assistidos pelos usuários.

32 Existem duas abordagens principais para a construção de sistemas de  
33 recomendação, como descrito na seção anterior. O primeiro tipo, chamado  
34 de filtragem colaborativa, demanda o uso de informações provenientes dos  
35 usuários. Desta forma, espera-se analisar informações comportamentais dos  
36 usuários como os filmes já assistidos, ratings dados pelo usuários e outras  
37 informações que permitam analisar similaridade entre os usuários. Já os sis-  
38 temas de recomendação com base no conteúdo tem como objetivo analisar  
39 as características dos filmes como, por exemplo, categorias, tags, diretores,  
40 elenco, relação entre as categorias e etc. Desta forma, como descrito anteri-  
41 ormente, o foco deste segundo tipo de abordagem está no conhecimento do  
42 filme que será recomendado. Uma terceira abordagem pode ainda combinar  
43 características das duas abordagens anteriores tendo uma solução mista.

44 O sistema de recomendação, como apresentado na imagem abaixo (Fig.  
45 1), tem como entrada um identificador único do usuário para o qual serão  
46 feitas recomendações de novos filmes. Na segunda etapa é construída a solu-  
47 ção de recomendação tendo como base as informações disponíveis no dataset.  
48 Estas informações podem ser sobre os filmes e/ou sobre os usuários. A so-  
49 lução pode seguir um dos três tipos de sistema de recomendação: filtragem  
50 colaborativa, sistemas baseados em conteúdo ou solução mista.

## 51 **3. ML-Latest-Small - Dataset**

52 O dataset [ml-latest-small] (<http://files.grouplens.org/datasets/movielens/ml-latest-small.zip>) que será utilizado neste projeto foi obtido no site [Grouplens]  
53 (<https://grouplens.org/>) mantido pelo Departamento de Ciência da Compu-  
54 tação e Engenharia da Universidade de Minnesota, EUA.

55 Este dataset é composto de informações obtidas no site [Movielens] (<https://movielens.org/>)  
56 que tem como objetivo recomendar filmes para os usuários. Existem duas  
57 versões do dataset. A primeira delas é destinada a pesquisas acadêmicas e  
58

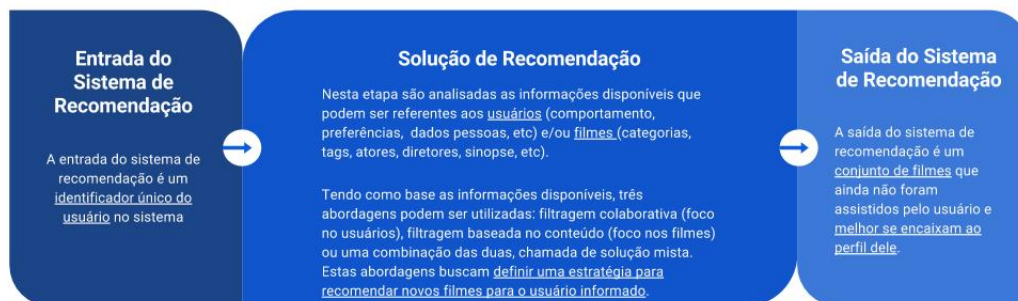


Figura 1: Visão Geral da Solução Proposta

59 a segunda para desenvolvimento e ensino. Por uma questão de poder com-  
60 putacional, será utilizada a versão reduzida do dataset destinado à ensino.  
61 Este é composto de aproximadamente 100 mil ratings, 1300 tags, 9 mil filmes  
62 avaliados por 700 usuários.

### 63 3.1. Dados Disponíveis

64 Para formular uma solução para o problema devemos analisar as infor-  
65 mações disponíveis no dataset a partir do qual o sistema de recomendação  
66 será desenvolvido. O conjunto de dados disponíveis neste dataset estão or-  
67 ganizados em 4 tabelas (.csv): links, movies, ratings, e tags. As features  
68 presentes em cada uma das tabelas são:

#### 69 3.1.1. Filmes

- 70 • Features: movieId, title, genres
- 71 • Total de 9125 filmes
- 72 • Amostra dos dados:

|    | movieId | title                              | genres    |
|----|---------|------------------------------------|-----------|
|    | 1       | Toy Story (1995)                   | Adventure |
|    | 2       | Jumanji (1995)                     | Adventure |
| 73 | 3       | Grumpier Old Men (1995)            | Comedy    |
|    | 4       | Waiting to Exhale (1995)           | Comedy    |
|    | 5       | Father of the Bride Part II (1995) | Comedy    |

#### 74 3.1.2. Ratings

- 75 • Features: userId, movieId, rating, timestamp;
- 76 • Total de 100.004 ratings
- 77 • Amostra dos dados:

|    | userId | movieId | rating | timestamp  |
|----|--------|---------|--------|------------|
|    | 1      | 31      | 2.5    | 1260759144 |
|    | 1      | 1029    | 3.0    | 1260759179 |
| 78 | 1      | 1061    | 3.0    | 1260759182 |
|    | 1      | 1129    | 2.0    | 1260759185 |
|    | 1      | 1172    | 4.0    | 1260759205 |

#### 79 3.1.3. Tags

- 80 • Features: userId, movieId, tag, timestamp;
- 81 • Total de 1296 tags
- 82 • Amostras dos dados:

|    | userId | movieId | tag                     | timestamp  |
|----|--------|---------|-------------------------|------------|
|    | 15     | 339     | sandra 'boring' bullock | 1138537770 |
|    | 1      | 1029    | dentist                 | 1193435061 |
| 83 | 15     | 7478    | Cambodia                | 1170560997 |
|    | 15     | 32892   | Russian                 | 1170626366 |
|    | 15     | 34162   | forgettable             | 1141391765 |

#### 84 3.1.4. Links

- 85 • Features: movieId, imdbId, tmdbId;
- 86 • Total de 9125 links
- 87 • Amostra dos dados:

|    | movieId | imdbId | tmdbId  |
|----|---------|--------|---------|
|    | 1       | 114709 | 862.0   |
|    | 2       | 113497 | 8844.0  |
| 88 | 3       | 113228 | 15602.0 |
|    | 4       | 114885 | 31357.0 |
|    | 5       | 113041 | 11862.0 |

### 89 3.2. Visualização Exploratória

90 Nesta seção são apresentadas informações extraídas durante a fase explo-  
 91 ratória dos dados. Nesta etapa informações como a investigação da correlação  
 92 entre variáveis e a predominância de alguma característica são buscadas. A  
 93 fase exploratória é fundamental para o conhecimento aprofundado dos dados  
 94 e a aplicação da técnica mais apropriada para resolver o problema proposta  
 95 de recomendação de filmes.

#### 96 3.2.1. Número de ratings por filme

97 A primeira avaliação feita diz respeito a distribuição de ratings por filmes.  
 98 Isso se deve ao fato de que é importante verificar quantos filmes possuem ava-  
 99 liações suficientes para que o sistema de recomendação possa ser construído.  
 100 Filmes com poucas avaliações são removidos do dataset, pois não apresentam  
 101 informações suficientes para construção do modelo. Na Figura 2 é possível  
 102 observar que a maioria dos filmes possuem entre 0 e 1 avaliação.

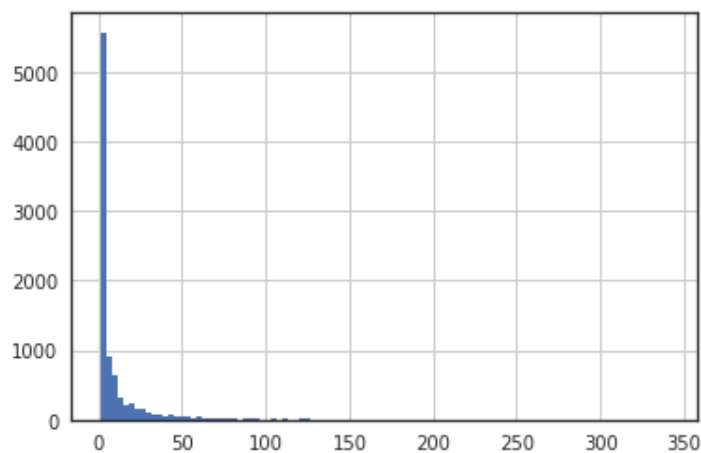


Figura 2: Histograma com a distribuição do número de ratings por filme.

103 A presença de dados suficientes para construção do modelo é determi-  
 104 nante para a decisão de não trabalhar com todos os dados existentes no  
 105 dataset. Desta forma, na Figura 3 é apresentado o gráfico após a remoção  
 106 dos filmes que apresentam menos de 4 ratings. É possível observar que mui-  
 107 tos dados foram filtrados do dataset. Mais precisamente, foram removidos  
 108 do dataset 5020 filmes dos 9066 iniciais (equivalente a 55% do dataset). A  
 109 seguir são apresentadas estatísticas do dataset inicial e do dataset filtrado,  
 110 respectivamente:

|     | métrica                    | dataset original | filtrado |
|-----|----------------------------|------------------|----------|
|     | número de ratings          | 9066             | 4046     |
|     | média de ratings por filme | 11               | 22       |
|     | desvio padrão              | 24               | 32       |
| 111 | número mínimo de ratings   | 1                | 4        |
|     | 25%                        | 1                | 6        |
|     | 50%                        | 3                | 11       |
|     | 75%                        | 9                | 25       |
|     | número máximo de ratings   | 341              | 341      |

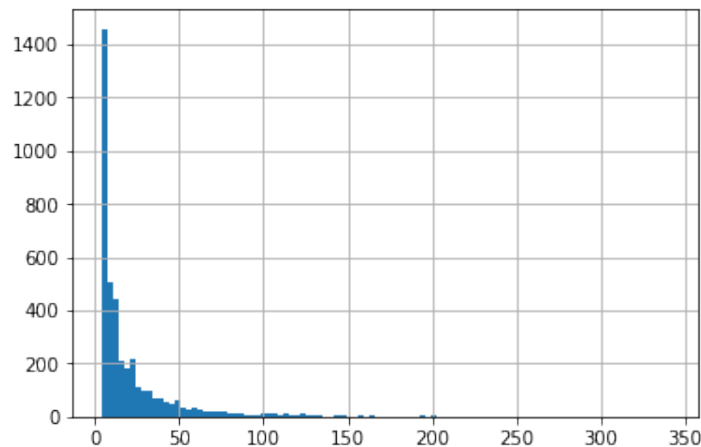


Figura 3: Histograma da distribuição de ratings por filme removendo aqueles que apresentam menos de 4 ratings.

### 112 3.2.2. Correlação entre ratings e categorias

113 Conhecer a correlação entre as variáveis do modelo é fundamental para  
 114 a construção da solução de recomendação. Desta forma, no Gráfico 4 é

115 possível observar a correlação entre o rating dado pelo usuário e as categorias  
 116 dos filmes presentes no dataset. Algumas categorias, como drama e crime  
 117 possuem uma correlação maior com os ratings que as demais (destacados em  
 118 vermelho).

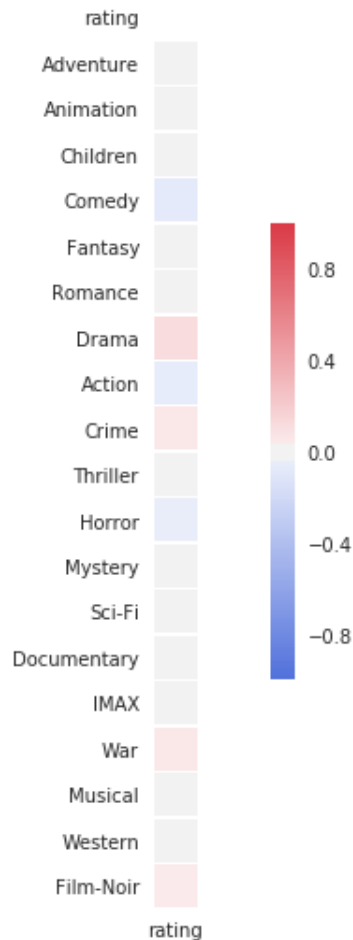


Figura 4: Correlação entre categorias e rating fornecido pelos usuários.

### 119 3.2.3. Correlação entre categorias

120 A correlação entre as categorias também é fundamental para a construção  
 121 da solução. É espero que exista correlação entre categorias como "Anima-  
 122 ção" e "Infantil" ou "Ação" e "Aventura". O Gráfico 5 apresenta estas cor-  
 123 relações em uma escala de cor entre vermelho (correlação positiva) e azul

(correlação negativa). A maior correlação é apresentada entre filmes infantis e animações e fantasia. Estas categorias claramente estão muito próximas, sendo bastante comum que filmes compartilhem mais de uma delas. Por outro lado, existem filmes com correlação negativa, como drama e ação, que são categorias bem diferentes. Estas correlações entre as categorias são fundamentais para a construção da solução proposta, apresentada na Seção 4.

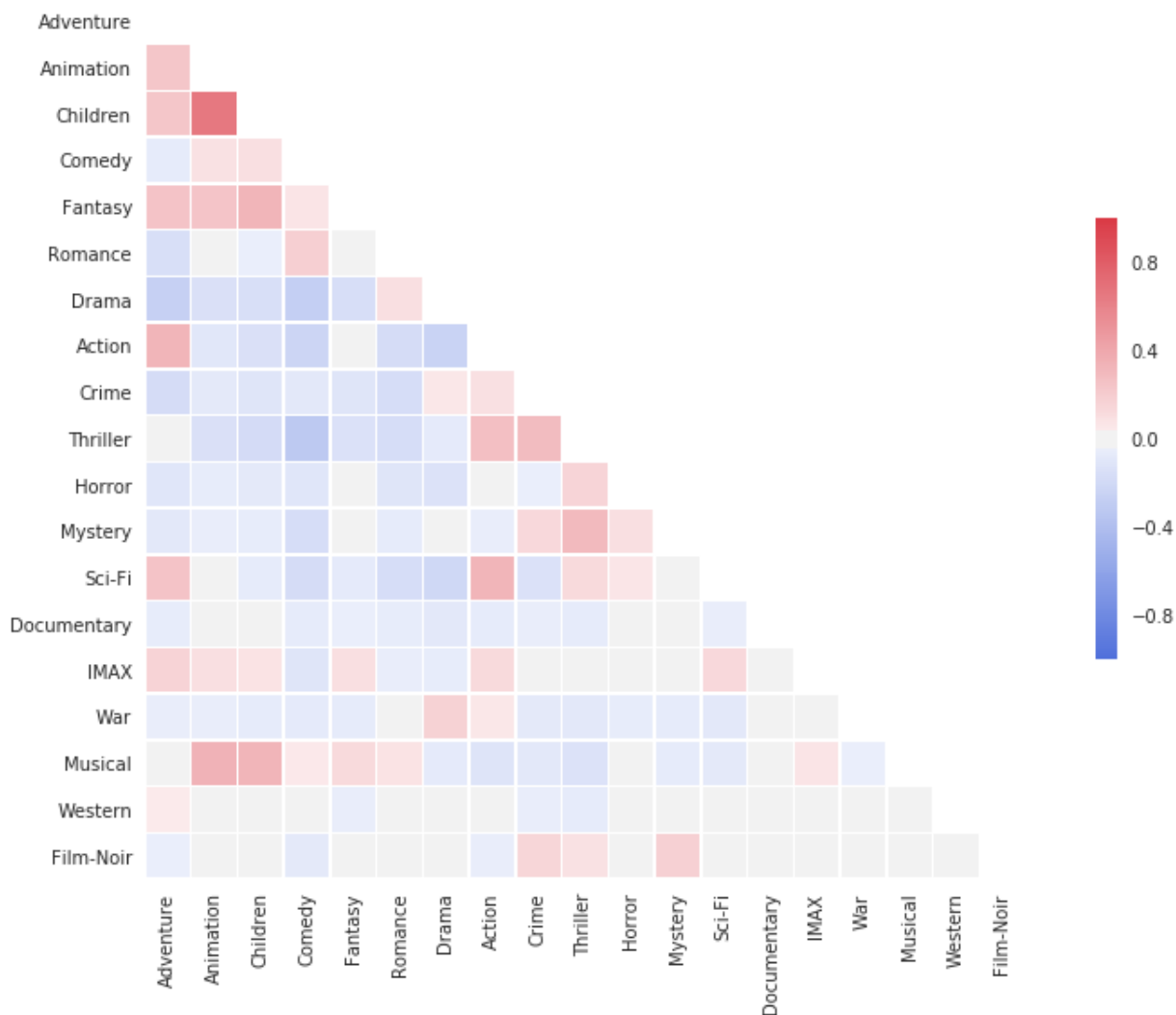


Figura 5: Correlação entre categorias.



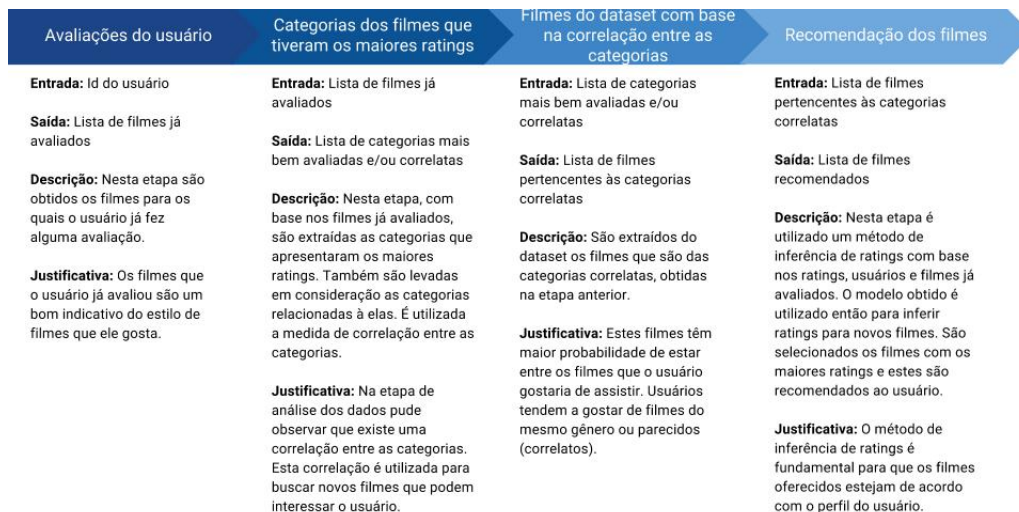


Figura 6: Visão Geral da Solução Proposta

## 130 4. Solução Proposta

131 A solução proposta para este projeto leva em consideração as informações  
 132 disponíveis no dataset [ml-latest-small] ([http://files.grouplens.org/datasets/movielens/ml-](http://files.grouplens.org/datasets/movielens/ml-latest-small.zip)  
 133 [latest-small.zip](http://files.grouplens.org/datasets/movielens/ml-latest-small.zip)), como apresentado na seção anterior.

134 O método proposto para o sistema de recomendação de filmes consiste em  
 135 4 etapas que compreendem desde a análise dos filmes assistidos pelo usuário  
 136 até a saída com uma lista de filmes recomendados. A Figura 6 apresenta  
 137 uma visão geral da solução.

138 A primeira etapa da solução, chamada de "Avaliação do usuário", con-  
 139 siste em processar a lista de filmes que o usuário já assistiu e avaliou. Esta  
 140 lista representa um bom indicativo do perfil de filmes que estão dentro das  
 141 preferências do usuário. Dentre os filmes avaliados, podemos inferir tam-  
 142 bém aqueles estilo que o usuário não gosta, representados pelos ratings mais  
 143 baixos dados por este usuário.

144 A segunda etapa consiste em avaliar a lista de filmes obtida na primeira  
 145 etapa, extraindo as categorias que foram mais bem avaliadas. Além disto, é  
 146 medida a correlação entre as categorias. As categorias com alta correlação são  
 147 levadas em consideração também. Esta solução tem como foco as categorias  
 148 dos filmes e suas correlações. A saída desta segunda etapa corresponde a

149 uma lista de categorias que representam o perfil do usuário. Vale ressaltar  
150 que as correlações entre as categorias devem ser medidas e validadas.

151 Na terceira etapa são obtidos os filmes que pertencem a estas categorias.  
152 São removidos os filmes que já foram assistidos pelo usuário. A justificativa  
153 principal para conduzir a busca deste filmes com base nas categorias é que os  
154 usuários tendem a gostar de filmes de uma mesmo estilo. Em outras palavras,  
155 os usuários tendem a gostar de filmes correlatos.

156 Na última etapa, para cada um dos filmes obtidos na etapa anterior, é  
157 utilizado um método de inferência de ratings. O modelo utilizado para a  
158 inferência é treinado utilizando os ratings, filmes e usuários pertencentes ao  
159 dataset. Tendo como base os ratings inferidos para o usuário, são seleciona-  
160 dos aqueles filmes com a maior pontuação. Por fim, estes filmes são então  
161 sugeridos para o usuário de entrada.

## 162 5. Resultados

### 163 5.1. Métricas

164 O método proposto será avaliado com base no modelo treinado para infe-  
165 rência de ratings, como apresentado na seção anterior. As métricas utilizadas  
166 serão:

#### 167 5.1.1. Root-mean-square error (RMSE)

$$RMSE = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - y^*_j)^2}$$

onde: número de elementos:

$$n$$

valor observado:

$$y_j$$

valor estimado:

$$y^*_j$$

|    | Agoritimo     | RMSE  | MAE   |
|----|---------------|-------|-------|
| 1  | SVD           | 0.896 | 0.69  |
| 2  | SVD++         | 0.886 | 0.68  |
| 3  | NMF           | 0.946 | 0.727 |
| 4  | Slope One     | 0.928 | 0.711 |
| 5  | k-NN          | 0.968 | 0.744 |
| 6  | Centered k-NN | 0.919 | 0.703 |
| 7  | k-NN Baseline | 0.896 | 0.686 |
| 8  | Co-Clustering | 0.965 | 0.749 |
| 9  | Baseline      | 0.893 | 0.69  |
| 10 | Random        | 1.44  | 1.151 |

Tabela 1: Benchmark para análise do modelo de predição de ratings. Fonte: <https://github.com/NicolasHug/Surprise/blob/master/examples/benchmark.py>

### 168 5.1.2. Mean Absolute Error (MAE)

$$MAE = \frac{1}{n} \sum_{j=1}^n |y_j - y^*_j|$$

onde: - número de elementos:

$$n$$

- valor observado:

$$y_j$$

- valor estimado:

$$y^*_j$$

### 169 5.2. Benchmarks

170 A avaliação do método proposto é fundamental para verificar quão boas  
171 estão sendo feitas as recomendações dos filmes. Existem diversos benchmarks  
172 que podem ser utilizados para essa finalidade:

173 - Surprise Scikit](<https://pypi.python.org/pypi/scikit-surprise>) - [My Me-  
174 dia Lite](<http://mymedialite.net/examples/datasets.html>)

175 Para a validação dos resultados foi executado o benchmark apresentado no  
176 repositório git: <https://github.com/NicolasHug/Surprise/blob/master/examples/benchmark.py>.  
177 Foi utilizado este benchmark executado para o dataset deste trabalho, como  
178 apresentado na tabela a seguir:

### 179 5.3. Conclusões

180 O trabalho desenvolvido tem como objetivo a construção, elaboração crítica e avaliação de um sistema de recomendação de filmes. Este tema bastante  
181 explorado na literatura foi abordado com foco nos sistemas de recomendação  
182 baseados em conteúdo.  
183

184 Os resultados são avaliados principalmente levando em consideração as  
185 métricas: Root-mean-square error (RMSE) e Mean Absolute Error (MAE),  
186 apresentadas nas seções anteriores. Estas métricas avaliam a qualidade do  
187 modelo construído na quarta etapa do método proposto (Seção 4) para a  
188 inferência de ratings. Verificou-se que, neste contexto, o ideal para a solução  
189 é utilizar todos os dados para a predição dos ratings e, posteriormente, para  
190 a seleção dos filmes utilizar a lista de filmes filtrada. Além disso, foi utilizada  
191 a validação cruzada (cross-validation) para melhor o desempenho do modelo.

192 Existem diversos métodos para inferência de ratings. Neste trabalho  
193 foi utilizado o Single-value decomposition (SVD) implementado no pacote  
194 python para sistemas de recomendação Surprise ([1]). Este algoritmo é bastante  
195 utilizado para redução de dimensionalidade. No contexto dos sistemas  
196 de recomendação o SVD é utilizado para a decomposição de matrizes. Este  
197 algoritmo foi escolhido por ser o mais comum nesta área de recomendação  
198 e também pela natureza do dataset. O dataset foi particionado da seguinte  
199 forma: 75% das amostras para treinamento e 25% para teste. Utilizando  
200 estas configurações as métricas obtidas foram de: RMSE (0,897) e MAE  
201 (0,691). Na Tabela 5.2 é possível observar que o SVD tem o melhor desempenho  
202 dentre os algoritmos avaliados.

203 Inicialmente, o modelo de inferência de ratings, que representa uma das  
204 etapas do método, foi treinado com o dataset reduzido, como apresentado  
205 na Seção 3.2.1. Neste cenário, filmes com um número muito pequeno de  
206 ratings foram removidos do dataset. Esta alternativa não se mostrou muito  
207 eficiente, tendo o SVD um desempenho bem inferior ao obtido na solução  
208 final com as métricas de: RMSE (1,79) e MAE (1,39). Por conta disto, a  
209 solução final utiliza o modelo de inferência de ratings treinado com o dataset  
210 original. Acredita-se que, com a redução no número de amostras do dataset,  
211 o problema tenha se tornado mais difícil.

212 Por fim, de forma qualitativa, os filmes recomendados foram considerados  
213 coerentes com o dataset de treinamento. É possível observar que as categorias  
214 tem um peso grande na definição dos filmes, representando um ponto que  
215 pode ser melhorado em trabalhos futuros. Além disso, outras abordagens

216 podem ser utilizadas para busca de semelhança entre os filmes como, por  
217 exemplo, o uso das palavras-chave.

218 [1] N. Hug, Surprise, a Python library for recommender systems, 2017.