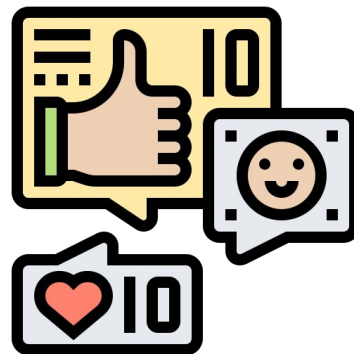




Programa de Pós Graduação em Ciências da Computação
Análise de Algoritmos e Estruturas de Dados - Profa. Dra. Lilian Berton

Recomendação de Sistemas Baseado em Grafos

Willian Dihanster Gomes de Oliveira



14 de Junho de 2021
São José dos Campos - SP

Introdução

- **Recomendação de Sistemas** é uma aplicação **bastante utilizada** em diversos contextos.
 - *Ecommerce, Streaming (Netflix, Spotify) e etc.*
- Diversas **abordagens e implementações**.

Nossas recomendações para você!

-79%	-33%	-55%	-53%
			
BOTA COTURNO TRATORADA ELEGAN...	SOBRETUDO PARKA RIOUTLET BENGALINE...	COTURNO AMORELLE PLATAFORMA SINTÉTIC...	BOTA CHELSEA COLCCI TIRAS PRETO
De 189,99 Por 39,99	De 150,00 Por 99,90	De 199,80 Por 89,90	De 319,90 Por 149,99
COMPRAR	COMPRAR	COMPRAR	COMPRAR

NETFLIX — Browse ▾

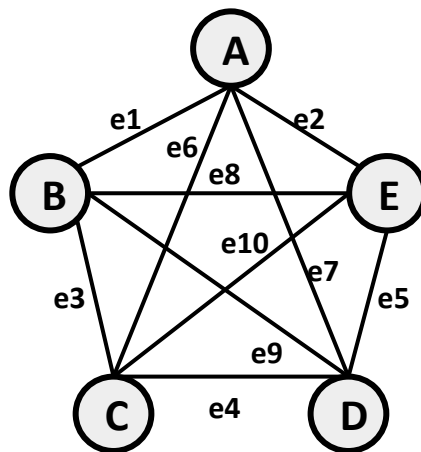
Recently Added

Because you added To Kill a Mockingbird to your list

Because you watched Helmut Schmidt – Lebensfragen

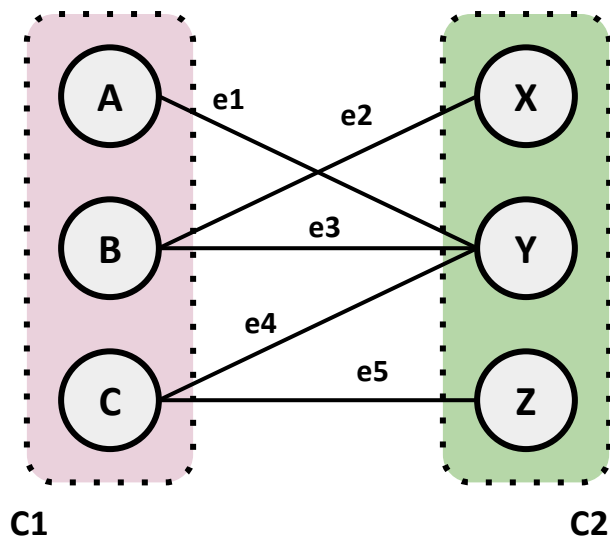
Grafos

- Tripla ordenada $G = (V(G), E(G), \Psi_G)$, em que $V(G)$ é um **conjunto de vértices** não-vazio, $E(G)$ um **conjunto de arestas** disjunto de $V(G)$, e Ψ_G , uma **função de incidência** que associa arestas a um par de vértices [6].
- Pode ser usado para **representar** diversos **relacionamentos**.
 - Exemplo: redes sociais, redes de comunicação e *etc.*



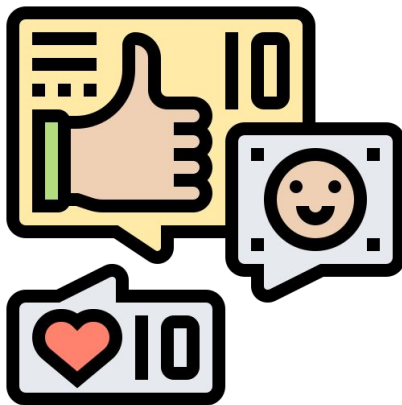
Grafos Bipartidos [1]

- Grafo que pode ser dividido em **dois conjuntos C1 e C2** [6]. Tal que:
 - Os dois **conjuntos** de vértices são **disjuntos**.
 - **Só existem arestas** de um **conjunto para o outro**.

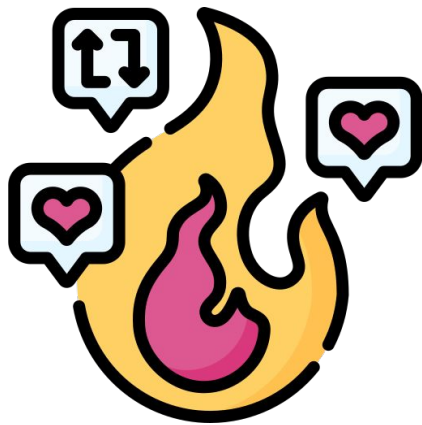


Recomendação de Sistemas [2]

- Programa que deve **prever/recomendar os itens** (produtos ou serviços) **mais adequados** para **determinado usuário**, com base em alguma heurística.
- Um tipo é a **filtragem colaborativa**:
 - Em *user-item*, recomenda-se itens que **usuários semelhantes** interagiram/gostaram.



Modelagem *Baselines*



***k*-Popularity**

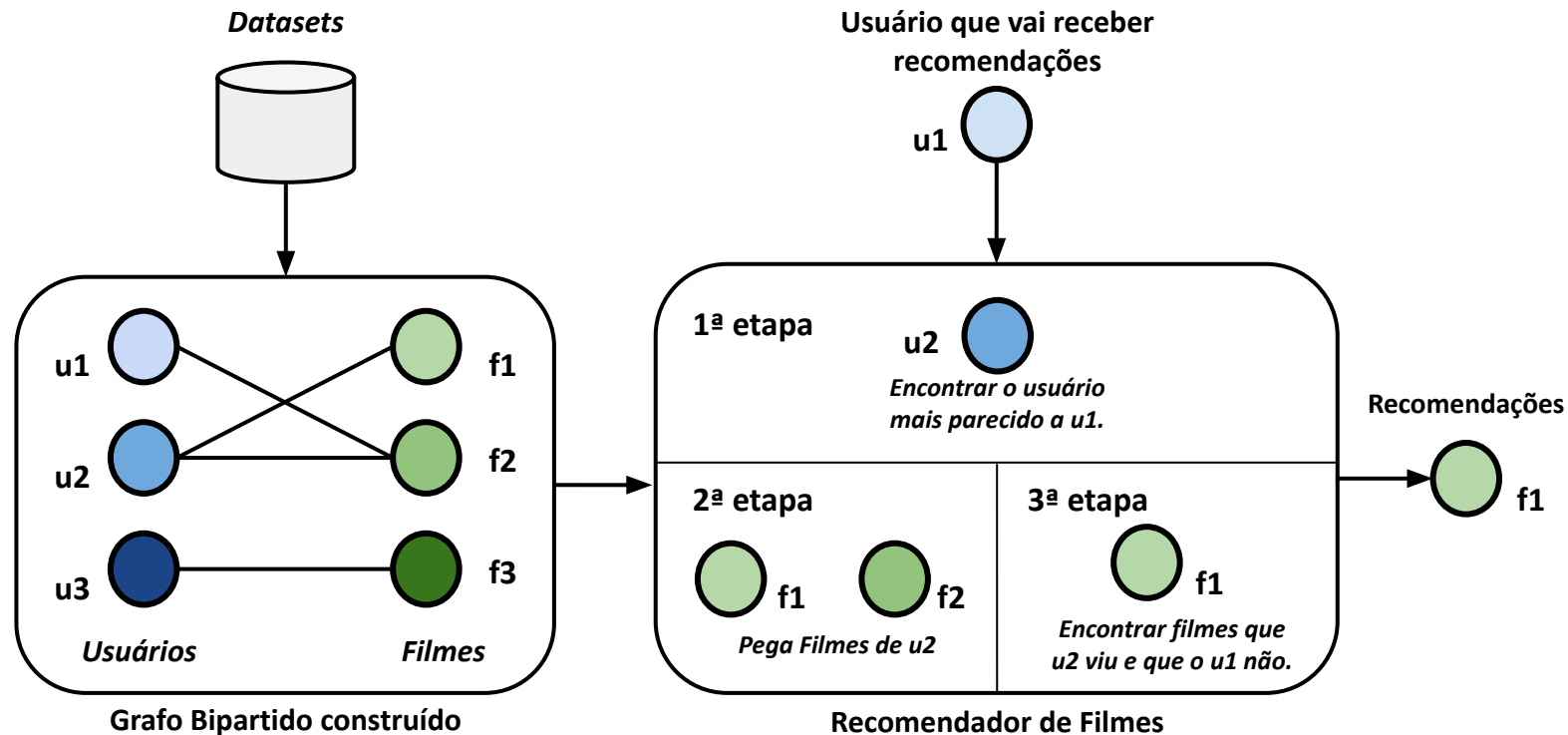
Recomenda os k itens mais populares



***k*-Random**

Recomenda k itens aleatoriamente

Modelo Baseado em Grafos [5]



Algoritmo e Complexidade

Algorithm 1: Função Principal do Algoritmo.

Data: G um grafo bipartido com U e F , conjuntos de usuários e filmes, respectivamente.

Result: R , dicionário de recomendações para cada Id.

begin

$R \leftarrow \emptyset$

$S \leftarrow \emptyset$

for $id \in G.userIds$ **do** $O(N)$

// Para cada usuário, devemos encontrar os usuários mais similares a ele.

$S \leftarrow mostSimilarUsers(G, id)$ $O(N*M)$

for $s \in S$ **do** $O(S)$

// Para cada usuário similar, devemos calcular os filmes a serem recomendados.

$moviesRecommended \leftarrow$
 $recommendMovies(G, id, s)$ $O(M)$

$R[id] \leftarrow moviesRecommended$

Passar por todos os nós e seus respectivos filmes.

Comparar filmes do usuário e filmes do seu similar.

$O(N) * (O(N*M) + O(S*M))$

$O(N) * O(N*M) = O(N^2*M)$

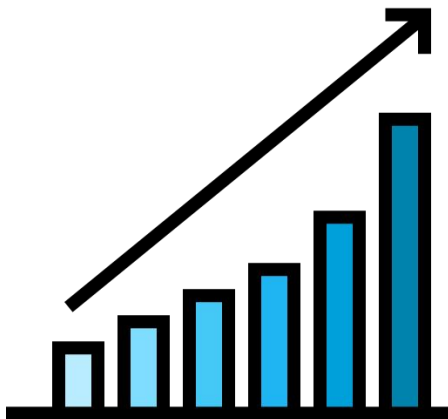
Experimentação - Dataset

- **Avaliação de usuários em filmes** do site [MovieLens](https://grouplens.org/datasets/movielens/), um serviço de recomendação de filmes.
- Dados de **671 usuários** e **9.123 filmes**. Além de **100.000 interações**.



Experimentação - Critérios de Análise

- **Coverage:** porcentagem dos filmes que o modelo consegue recomendar / “cobrir”.
- **Personalization:** índice de personalização, o quanto o modelo tá conseguindo ser personalizado para cada usuário. Isto é, não está recomendando a mesma coisa.



Resultados

$k = 10$

Métrica	Graph-based	Random	k -Popularity
Coverage	23,43	51,89	0,11
Personalization	0,9920	0,99889	0,0030
Tempo (em s)	9,82	6,69	3,28

$k = 30$

Métrica	Graph-based	Random	k -Popularity
Coverage	34,84	89,52	0,27
Personalization	0,9887	0,9967	0,0030
Tempo (em s)	10,32	7,15	3,36

$k = 50$

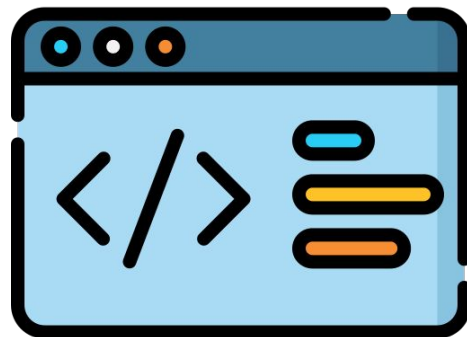
Métrica	Graph-based	Random	k -Popularity
Coverage	39,85	97,52	0,43
Personalization	0,9876	0,9945	0,0029
Tempo (em s)	9,95	7,21	3,94

k -Popularity pior. Esperado, pois, não oferece *coverage*, pois só cobre k produtos e nem *personalization*, pois recomenda os mesmos k para todos usuários.

Aleatório perfeito em *personalization*, porque suas recomendações são totalmente aleatórias, o que acaba sendo diferente para todos usuários e consegue cobrir (*coverage*) bastante filmes.

Graph-based melhor que o k -popularity, com bom *personalization* e com um *coverage* médio.

Demonstração



Conclusões

- Construímos um **recomendador baseado em grafos** com **bom *coverage* e *personalization***. Ao menos melhor que o modelo *k*-Popularity.
- **Não foi melhor** que o Random.
 - **Esperado** dado as **métricas** utilizadas que **beneficiam a aleatoriedade**.
 - ***Accuracy***, possivelmente muito **pior** para o **Random**, **que chuta as recomendações**.
- Testar **métricas mais robustas** para **cálculo de similaridade** entre usuários.

Referências

- [1] S. Sawant, “Collaborative filtering using weighted bipartite graph projection: a recommendation system for yelp,” in Proceedings of the CS224W: Social and information network analysis conference, vol. 33, 2013.
- [2] H. Wang and S. Chen, “A bipartite graph-based recommender for crowdfunding with sparse data,” in Banking and Finance. IntechOpen, 2020.
- [3] J. Wang, F. Shao, S. Wu, R. Sun, and R. Li, “Weighted bipartite network projection for personalized recommendations,” Journal of Advances in Computer Networks, vol. 4, no. 1, pp. 64–69, 2016.
- [4] B. Chen, J. Wang, Q. Huang, and T. Mei, “Personalized video recommendation through tripartite graph propagation,” in Proceedings of the 20th ACM international conference on Multimedia, 2012, pp. 1133–1136.
- [5] B. Shams and S. Haratizadeh, “Graph-based collaborative ranking,” Expert Systems with Applications, vol. 67, pp. 59–70, 2017.
- [6] J. A. Bondy, U. S. R. Murty et al., Graph theory with applications. Macmillan London, 1976, vol. 290.
- [7] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, “Recommender system application developments: a survey,” Decision Support Systems, vol. 74, pp. 12–32, 2015.
- [8] T. Zhou, J. Ren, M. Medo, and Y.-C. Zhang, “Bipartite network projection and personal recommendation,” Physical review E, vol. 76, no. 4, p. 046115, 2007.
- [9] F. Cacheda, V. Carneiro, D. Fernandez, and V. Formoso, “Comparison of ´ collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems,” ACM Transactions on the Web (TWEB), vol. 5, no. 1, pp. 1–33, 2011.

Referências (Código-fonte):

<https://github.com/kurasaiteja/Github-Recommender-System>

<https://github.com/statisticianinstilettos/recmetrics/blob/e8d9b39131999c484c4c98265f91e233be6ca4cb/recmetrics/metrics.py#L160>

Obrigado!