EPFL

Movinder: A Movie Recommendation System for Groups

Course:
A Network Tour of Data Science (EE-558)

Team:
#6

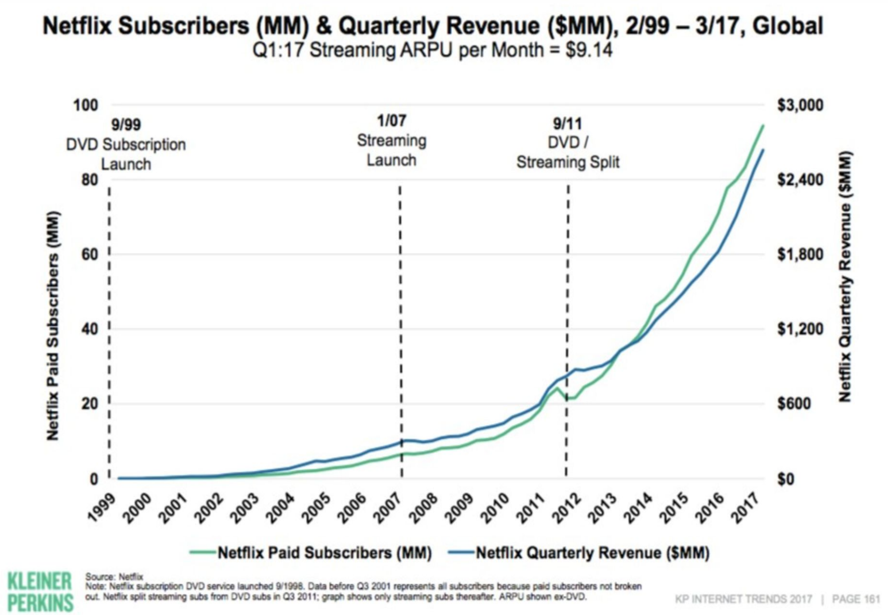
Lausanne, January 22nd 2020

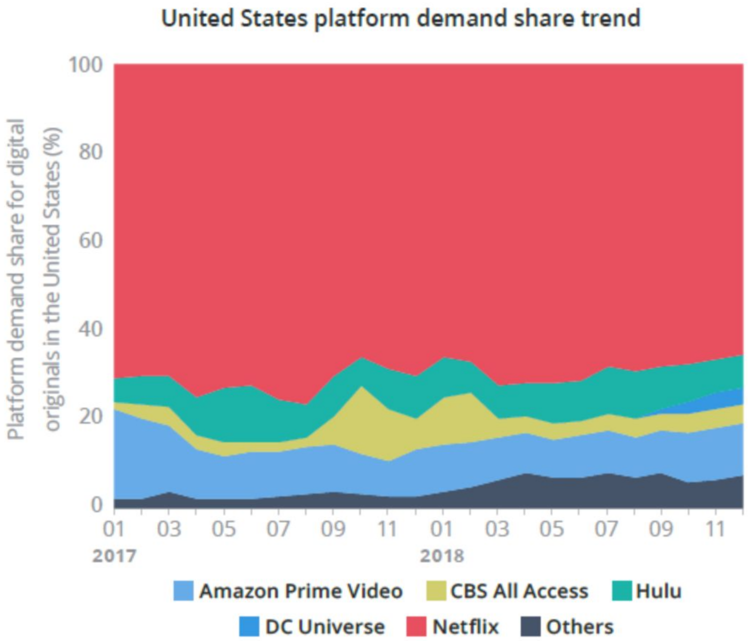
Content

1. Motivation
2. Acquisition
3. Exploration & Exploitation
4. Communication

1. Motivation

Movie Streaming: A Booming Industry



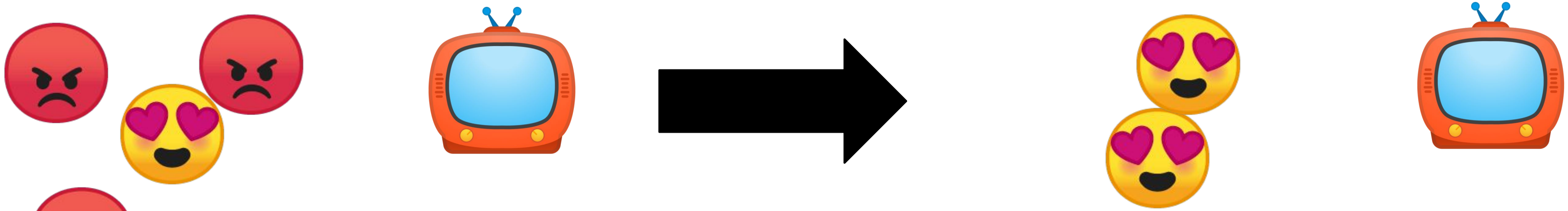


Source: indiewire.com

- Services compete in content & recommendations

The Case For Multi-User Recommendations

Recommender systems:
adapt to user preferences…
but satisfy only one person at a time



Socially-aware recommendations:
Aggregate the preferences of multiple users

- Movinder: graph-based multi-user recommender

1. Motivation

(skipped during presentation)

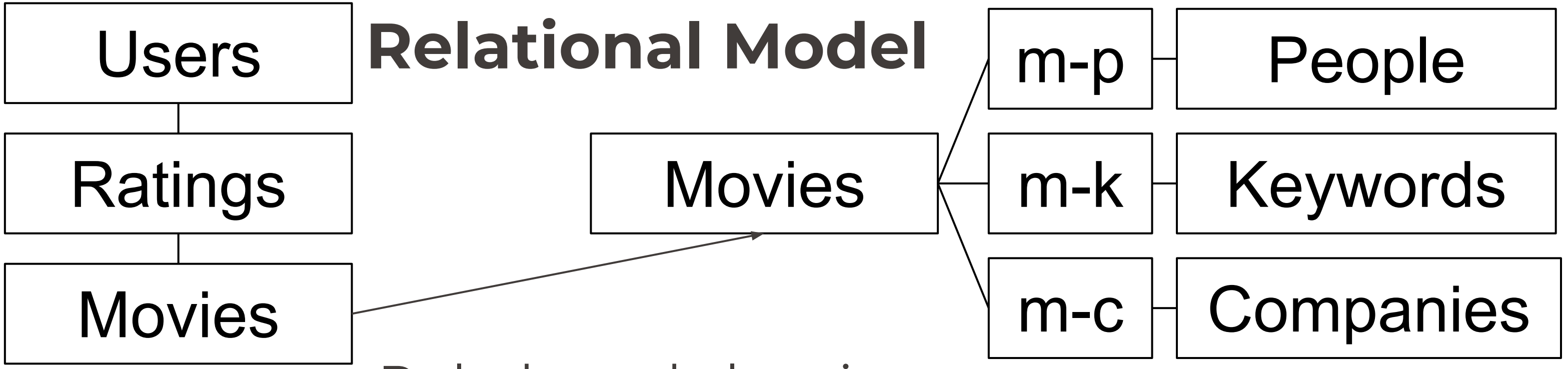
- Watching movies is a common activity to do with friends
- But deciding what to watch can take even hours and lead to disputes!

- Solution: Moviender!
- A multi-user graph recommendation system using data from **MovieLens** and **IMDb**

2. Acquisition

Data Acquisition

Data Collection



movielens

10000 ratings
943 users
1682 movies

Rule-based cleaning
Join on title, release date

Feature Vector
0-1 encoding

IMDb

134170 keywords
4167491 cast members
234997 companies
and more…

Composite dataset in vector representation

2. Acquisition

(skipped during presentation)

- Core dataset: MovieLens
  - 10000 ratings from 943 users on 1682 movies
  - Information about users (age, gender) and movies (genres, year of release)
- Extended with IMDb
  - Extra information on movies about keywords, cast members, production companies
- Rule-based data cleaning to merge the datasets

Building the Graph - Data Analysis

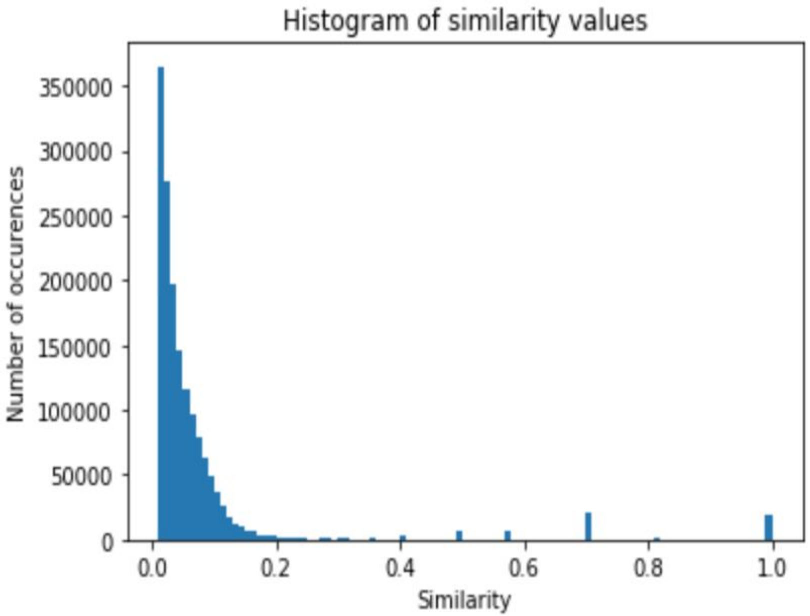
- Few very similar movies
- Not sparse

Feature Vector

Cosine-similarity
Sparsify with threshold

Graph

Nodes: movies
Edges: similarity



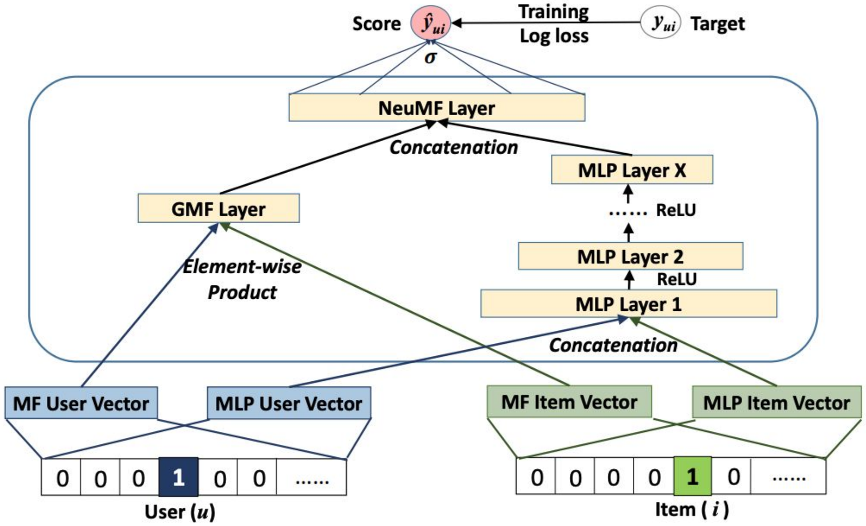
Cutoff Threshold: 0.1

- 22 nodes without neighbours
- Non-zero values: 6.98 - relatively sparse
- Hubs with more than 400 neighbors

3. Exploration & Exploitation

Recommendations with Experiments

Exploration & Exploitation
3.1 Neural Collaborative Filtering

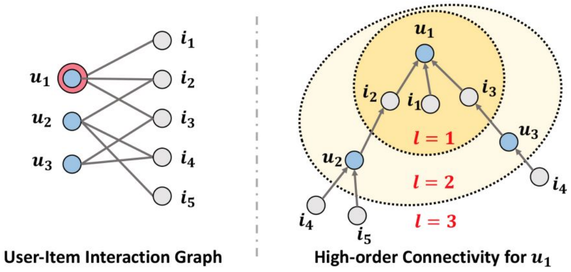


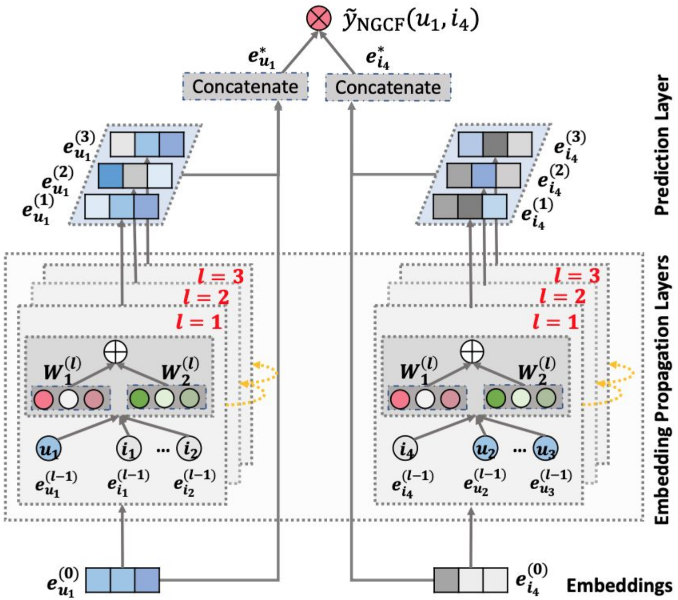
$$
\begin{aligned}
\phi^{GMF} &= \mathbf{p}_u^G \odot \mathbf{q}_i^G, \\
\phi^{MLP} &= a_L(\mathbf{W}_L^T(a_{L-1}(...a_2(\mathbf{W}_2^T \begin{bmatrix} \mathbf{p}_u^M \\ \mathbf{q}_i^M \end{bmatrix} + \mathbf{b}_2)...)) + \mathbf{b}_L), \\
\hat{y}_{ui} &= \sigma(\mathbf{h}^T \begin{bmatrix} \phi^{GMF} \\ \phi^{MLP} \end{bmatrix}),
\end{aligned}
$$

- NeuMF consists of two parts, General Matrix Factorization (GMF) and a second one which is Multi-Layer Perceptron (MLP).

- Grid Search w/ L1 Loss.

Exploration & Exploitation
3.2 Neural Graph Collaborative Filtering





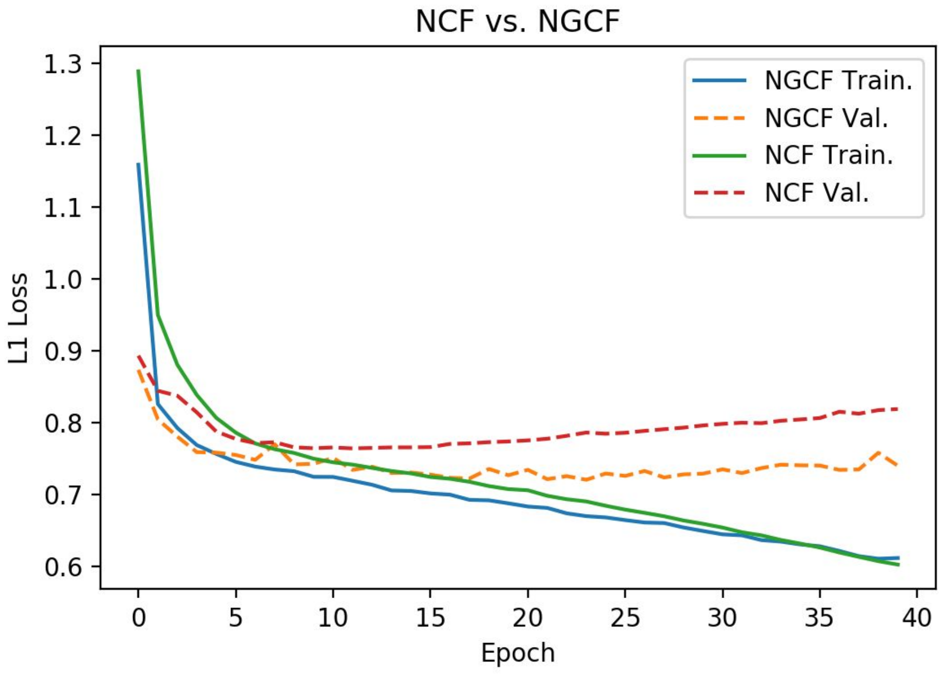
- Propagating embeddings recursively on the graph.

- Constructing information flows in the embedding space.

- *Embedding propagation layer refines a user's (or an item's) embedding by aggregating the embeddings of the interacted items (or users).*

Exploration & Exploitation

3.3 NGCF vs. NCF



NGCF Test Error:

0.72 MAE

NCF Test Error:

0.76 MAE

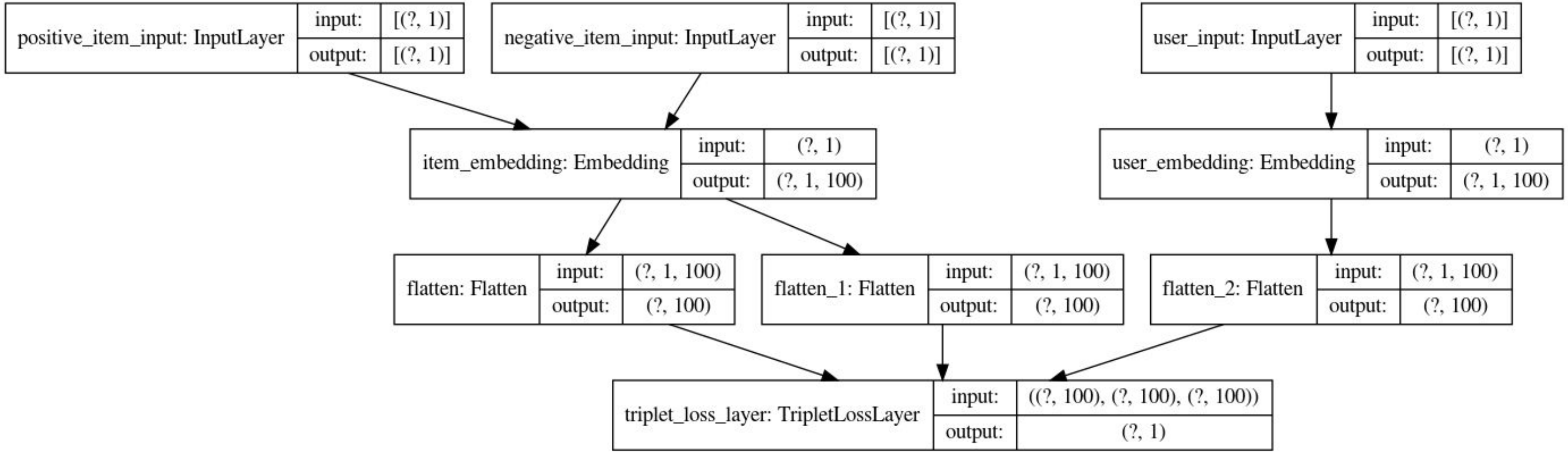
Both models

require GPU

For new user prediction

Exploration & Exploitation

3.4 Siamese Neural Network



$$L_{BPR}(a,p,n) = \sum 1 - \sigma(f(a,p) - f(a,n))$$

$L_{BPR}(a,p,n)$ - Bayesian Personalized Ranking (BPR) loss

a - anchor observation

p - positive sample close to a

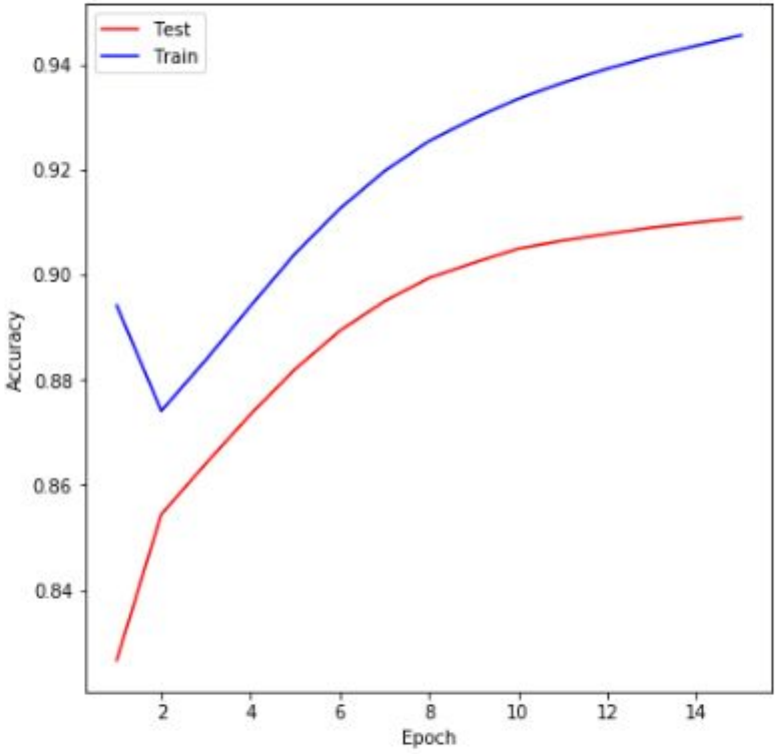
n - negative sample far from a

σ - sigmoid function

f - transformation we want to learn

Exploration & Exploitation

3.4 Siamese Neural Network



Training accuracy:

94.5%

Test accuracy:

91.1%

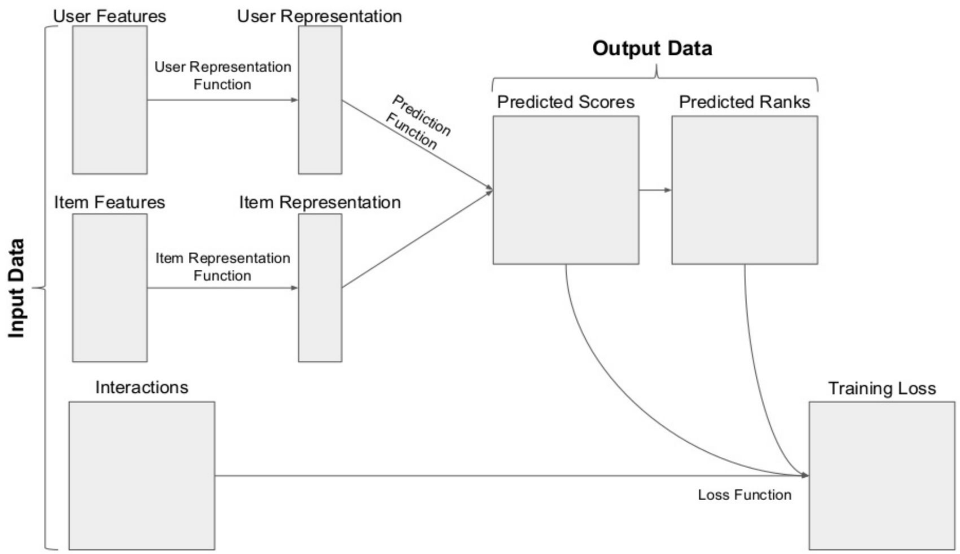
However,

~3-5 min

For new user prediction

Exploration & Exploitation

3.5 LightFM



Training accuracy:
87%

Test accuracy:
88%

~10-20 sec
for new user prediction

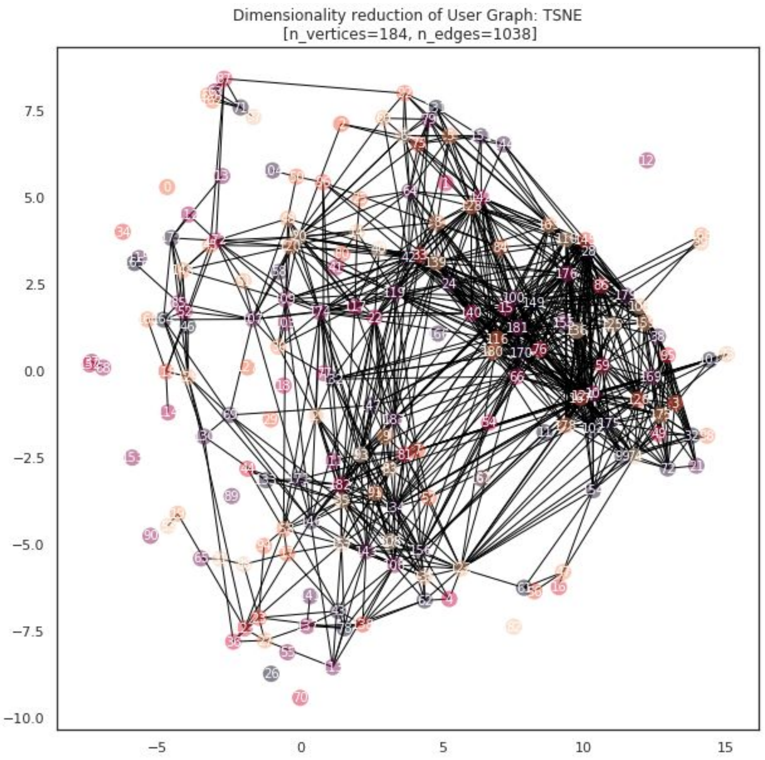
Settings:
- # of epochs: **150**
- Learning rate: **0.015**
- Ranking loss: **WARP* loss**
- etc.

*Weighted Approximate-Rank Pairwise

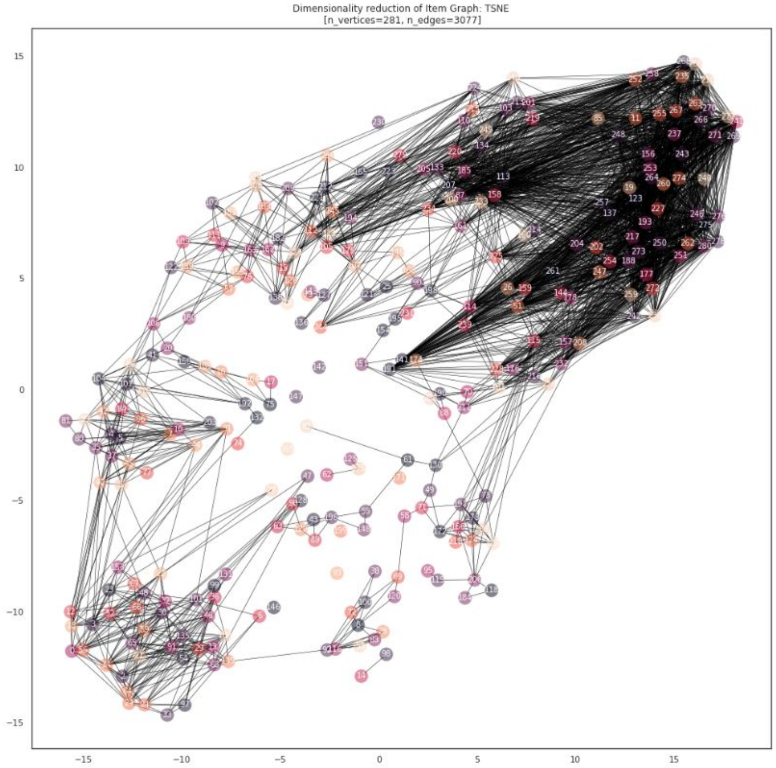
Exploration & Exploitation

3.5 LightFM

Friends Graph



Movie Graph



4. Communication

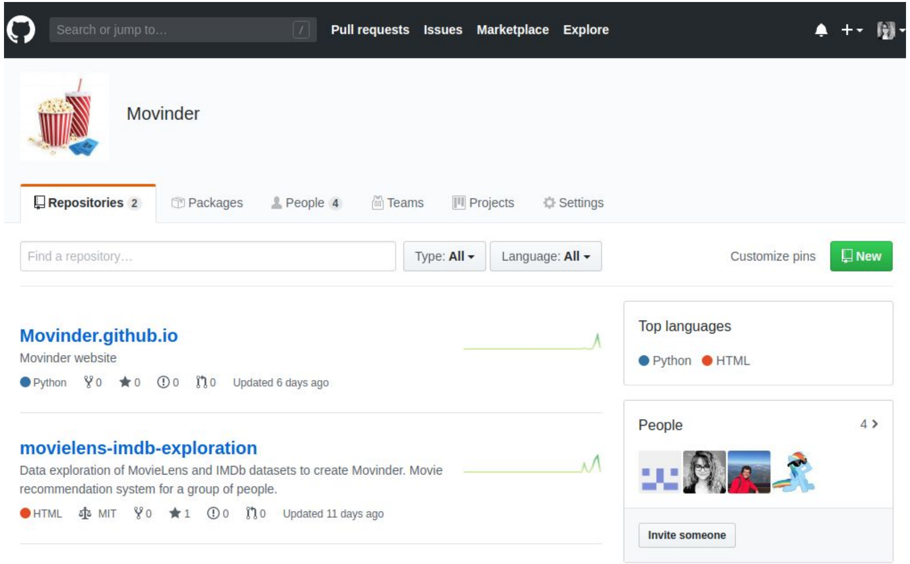
Website Product

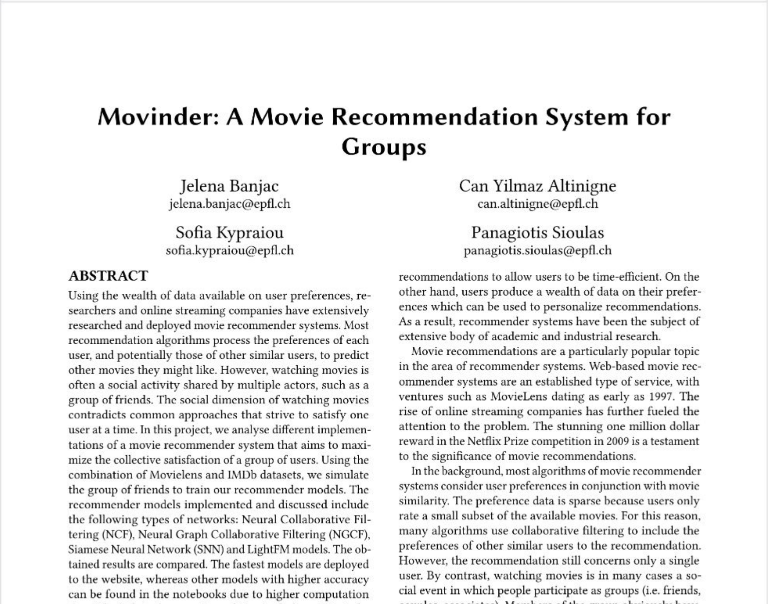
Communication

GitHub repository:

https://github.com/Movinder

Report





Movinder: A Movie Recommendation System for Groups

Jelena Banjac
jelena.banjac@epfl.ch

Can Yilmaz Altinigne
can.altinigne@epfl.ch

Sofia Kypraiou
sofia.kypraiou@epfl.ch

Panagiotis Sioulas
panagiotis.sioulas@epfl.ch

ABSTRACT

Using the wealth of data available on user preferences, researchers and online streaming companies have extensively researched and deployed movie recommender systems. Most recommendation algorithms process the preferences of each user, and potentially those of other similar users, to predict other movies they might like. However, watching movies is often a social activity shared by multiple actors, such as a group of friends. The social dimension of watching movies contradicts common approaches that strive to satisfy one user at a time. In this project, we analyse different implementations of a movie recommender system that aims to maximize the collective satisfaction of a group of users. Using the combination of Movielens and IMDb datasets, we simulate the group of friends to train our recommender models. The recommender models implemented and discussed include the following types of networks: Neural Collaborative Filtering (NCF), Neural Graph Collaborative Filtering (NGCF), Siamese Neural Network (SNN) and LightFM models. The obtained results are compared. The fastest models are deployed to the website, whereas other models with higher accuracy can be found in the notebooks due to higher computation

recommendations to allow users to be time-efficient. On the other hand, users produce a wealth of data on their preferences which can be used to personalize recommendations. As a result, recommender systems have been the subject of extensive body of academic and industrial research.

Movie recommendations are a particularly popular topic in the area of recommender systems. Web-based movie recommender systems are an established type of service, with ventures such as MovieLens dating as early as 1997. The rise of online streaming companies has further fueled the attention to the problem. The stunning one million dollar reward in the Netflix Prize competition in 2009 is a testament to the significance of movie recommendations.

In the background, most algorithms of movie recommender systems consider user preferences in conjunction with movie similarity. The preference data is sparse because users only rate a small subset of the available movies. For this reason, many algorithms use collaborative filtering to include the preferences of other similar users to the recommendation. However, the recommendation still concerns only a single user. By contrast, watching movies is in many cases a social event in which people participate as groups (i.e. friends,

Communication

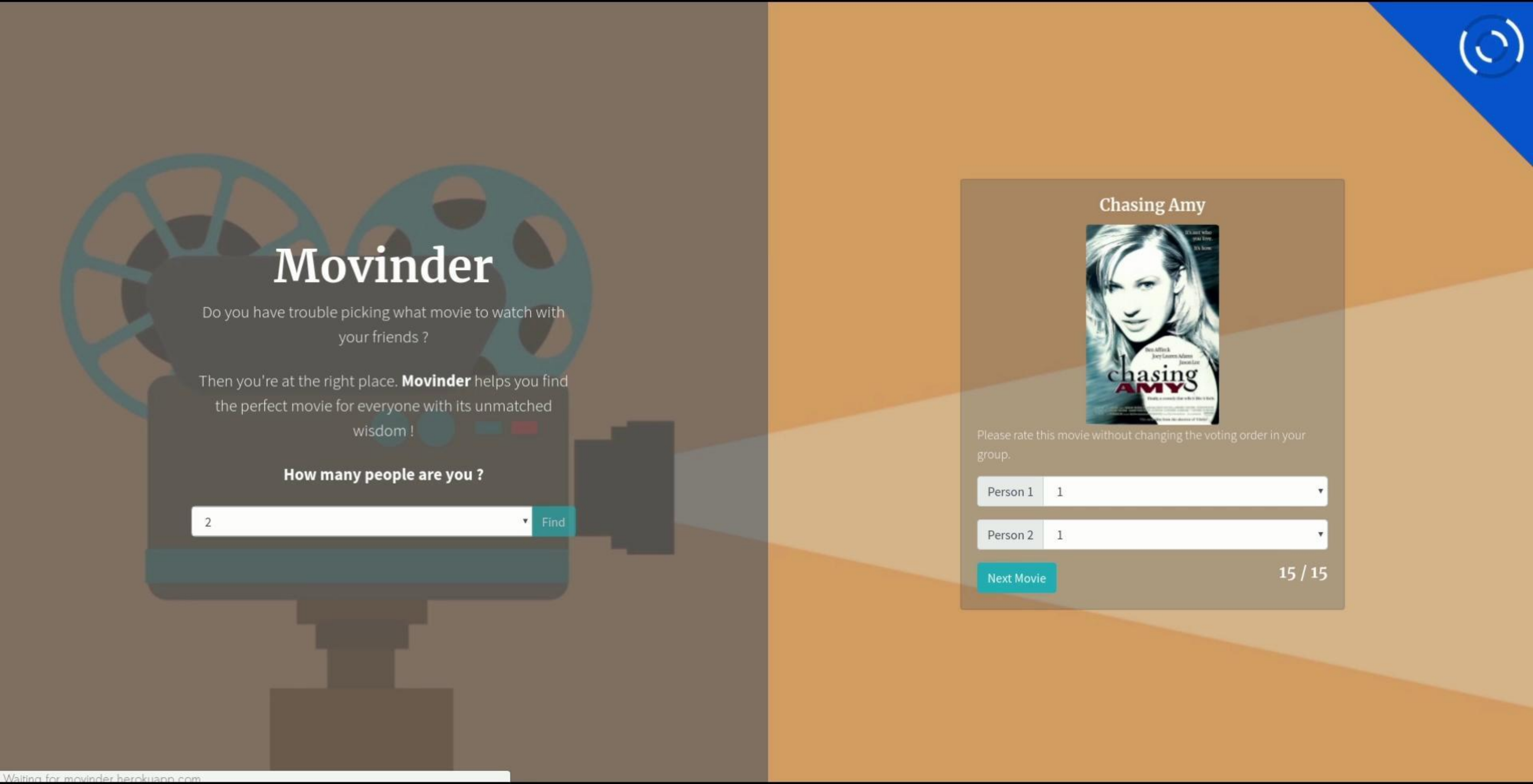
Website:

https://movinder.herokuapp.com/

Demo video:

https://youtu.be/zx0AxmEK05g

Next slide...


Movinder
Do you have trouble picking what movie to watch with your friends ?
Then you're at the right place. Movinder helps you find the perfect movie for everyone with its unmatched wisdom !
How many people are you ?
2
Find
Chasing Amy
chasing AMY
Please rate this movie without changing the voting order in your group.
Person 1
1
Person 2
1
Next Movie
15 / 15
Waiting for movinder.herokuapp.com

Thank you!

Questions?