Graph Based Recommendation

ANDREA MANZINI

LORENZO LAZZARA

MARTIN JOSIFOSKI

MAZEN FOUAD A-WALI MAHDI

Overview

- Description of the project
- Data Exploration
  - Listening count analysis
  - Friendship Network
  - Weight Normalization
  - Tag analysis
- Artist Similarity Network
  - Collective tagging
  - Retrieving artist similarity using the API
  - Artist Similarity Network construction
- Inferred User-User Network
- Recommender System

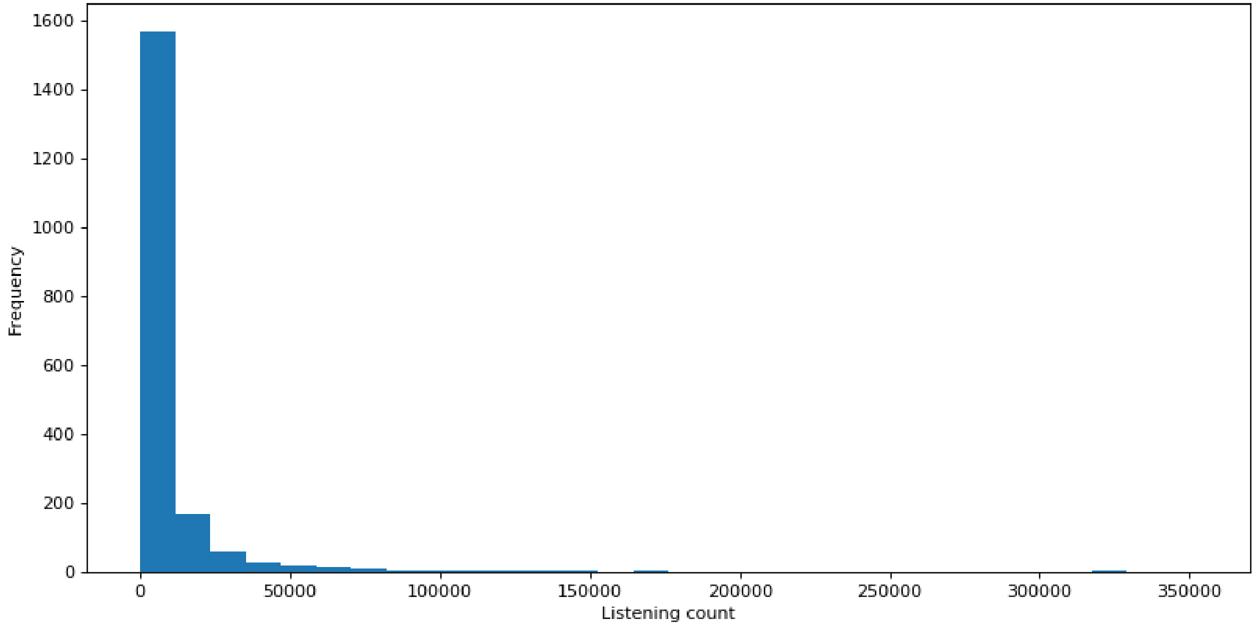
Description of the Project

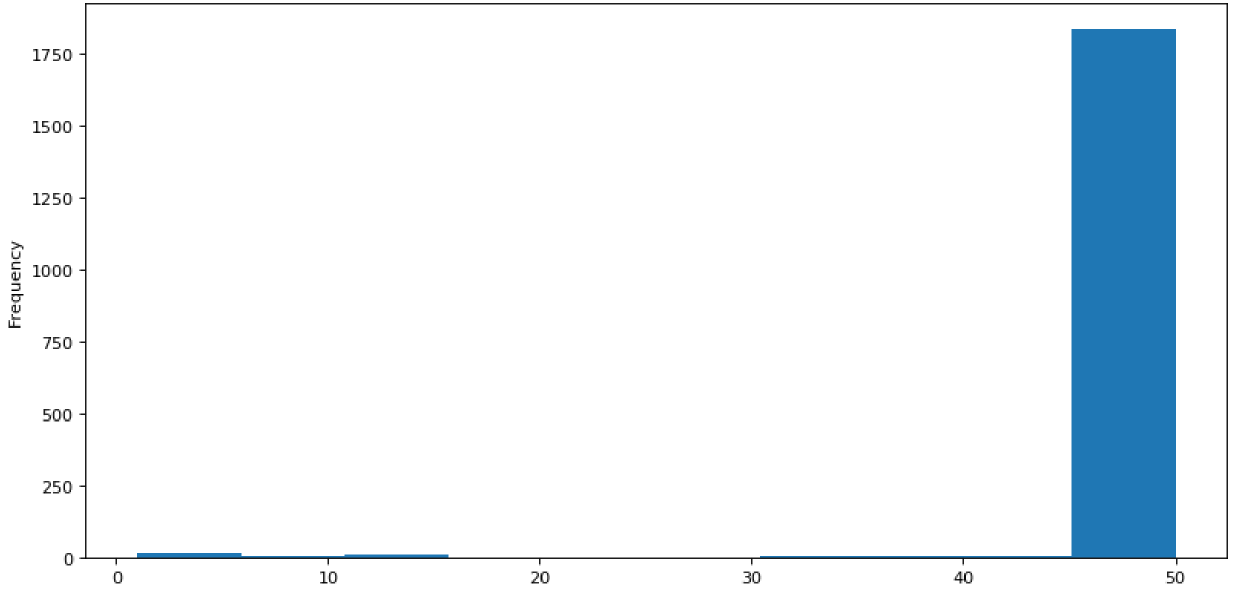
- Dataset from LastFM:
  - Social Network
  - Artist tags assigned by user
  - User-Artist listening counts
  - Set of 1892 users and 17632 artists
- Analyze the social network to understand if the friendship is based on musical taste
- Include in a recommendation system the informations from the artist similarity network or the social network and analyze the differences.

Data Exploration

Listening count analysis

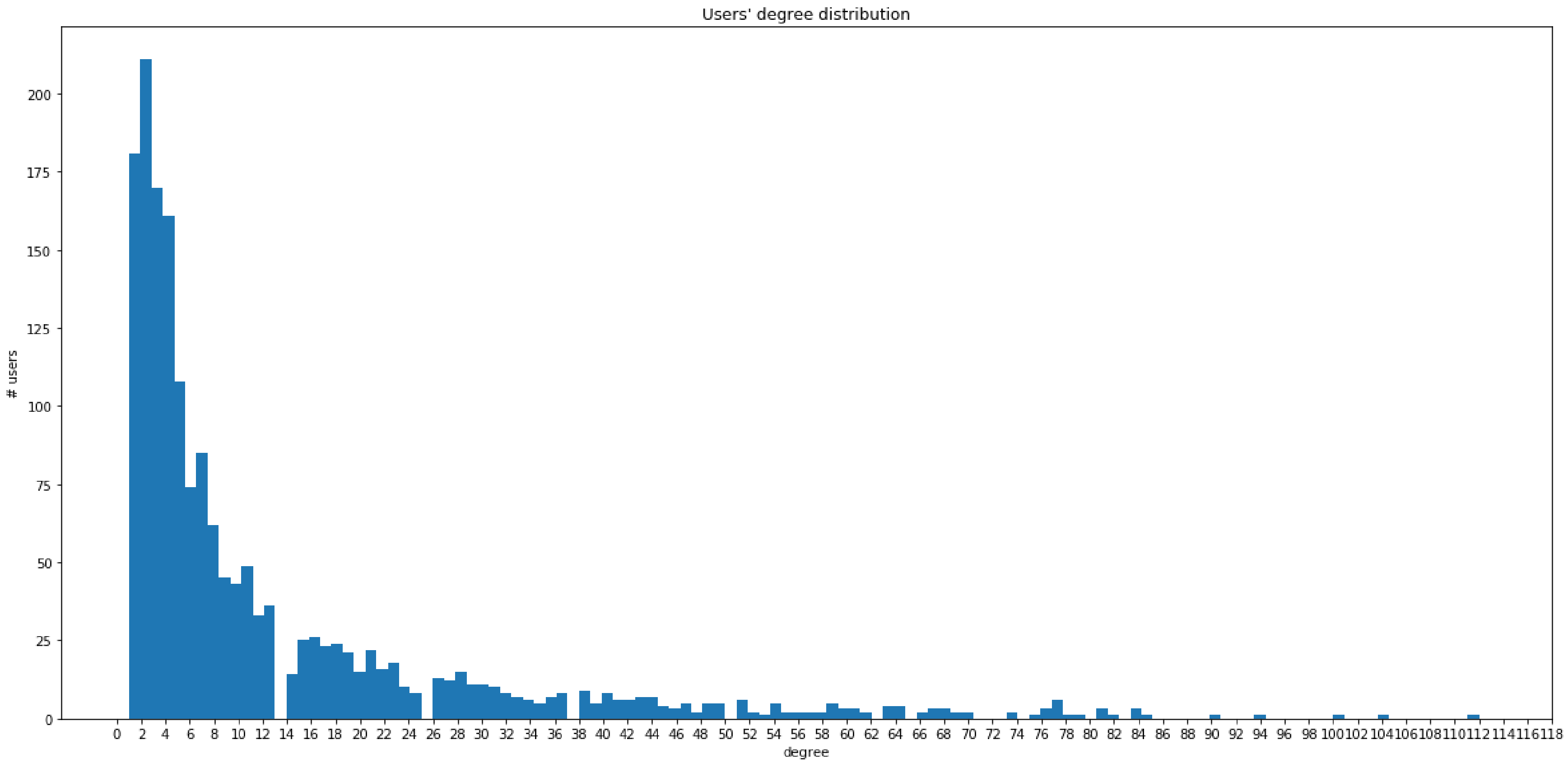
- Listening count: time (in seconds) a user spent listening to an artist.
- Prune users with maximum listening count higher than 50k.
- Prune users with less than 10 connected artists.





Friendship Network

- Friendship network without the users already pruned
- Network composed of several connected components.
- Consider only the giant component to obtain a connected graph.

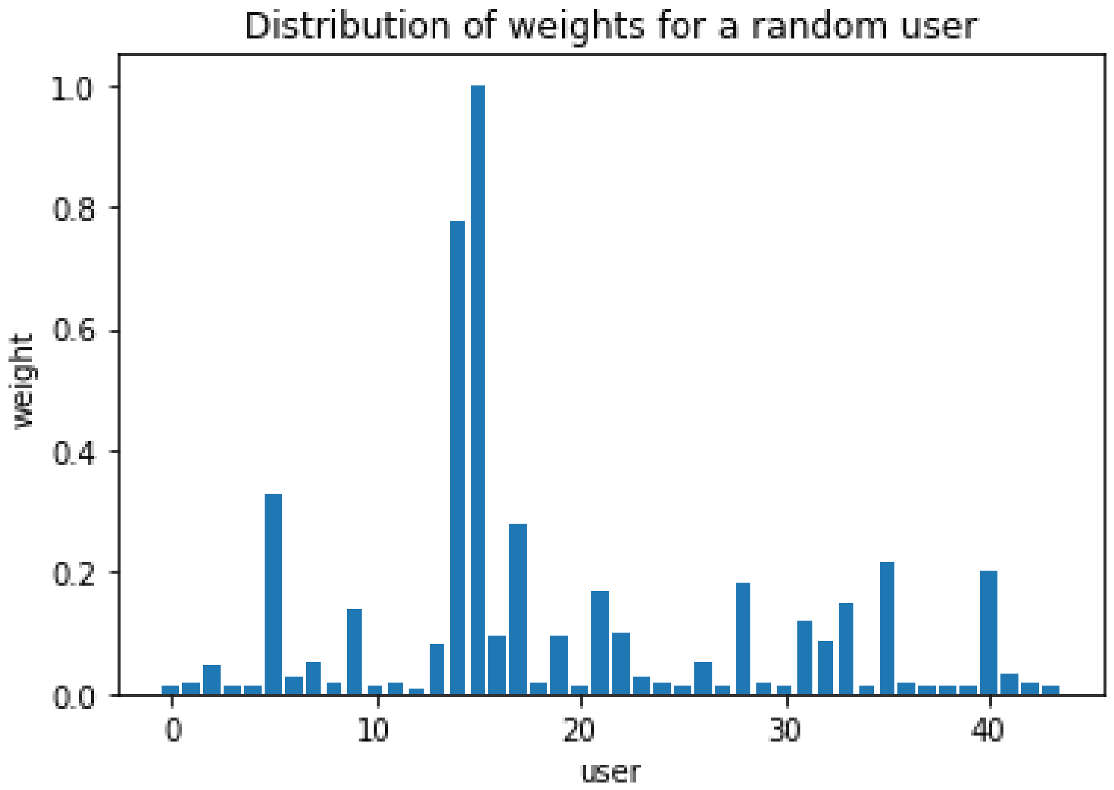


Weight Normalization

User dependent normalization:

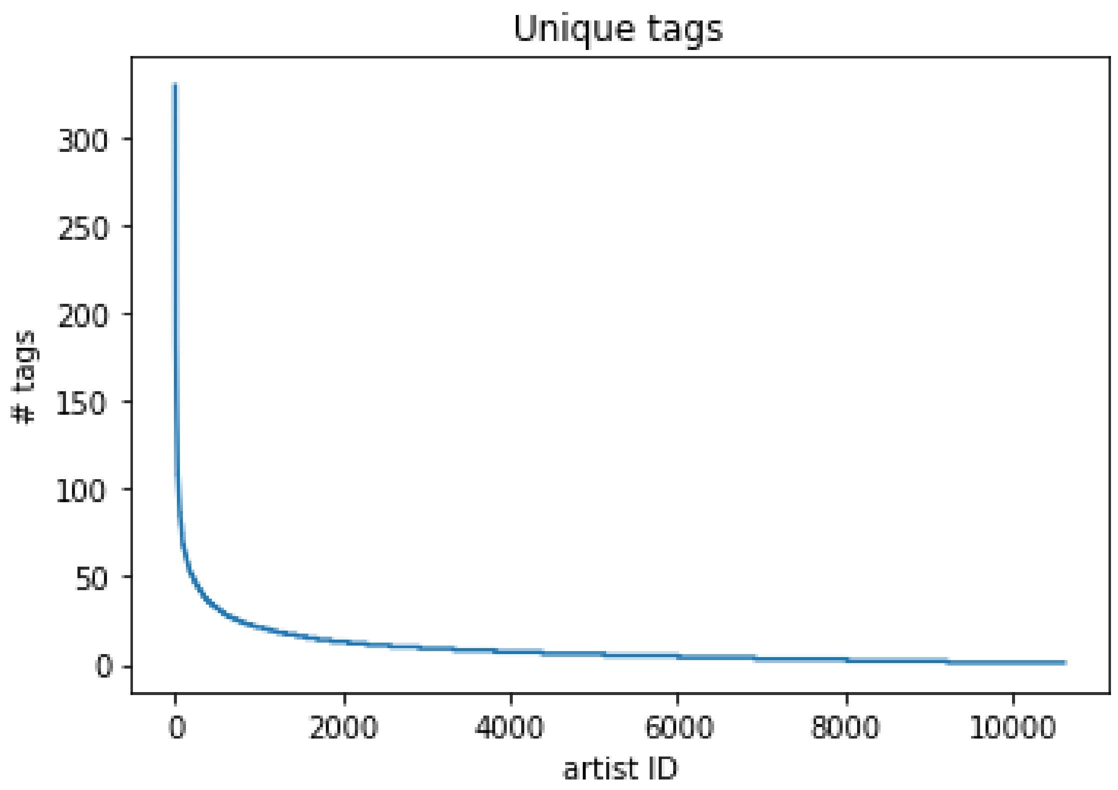
$$W_{ij} = \frac{L_{ij}}{\max(L_{i:})}$$

Where W is the weight, L is the listening count, i denote the user and j denote the artist



Tag analysis

- ❑ Freeform tags assigned by users to artists
- ❑ Possibility to use tags to identify artists’ musical genre
- ❑ Two main problems:
  - ❑ Meaningless tags
  - ❑ Different tags with the same meaning (e.g. 90’s -> 90 -> ninety)



Collective Tagging

- Mode of archiving.
- Ease of finding music.
- Helps in recommending music.

Using the Api

id	name	url	pictureURL
1	MALICE MIZER	http://www.last.fm/music/MALICE+MIZER	http://userserve-ak.last.fm/serve/252/10808.jpg
2	Diary of Dreams	http://www.last.fm/music/Diary+of+Dreams	http://userserve-ak.last.fm/serve/252/3052066.jpg
3	Carpathian Forest	http://www.last.fm/music/Carpathian+Forest	http://userserve-ak.last.fm/serve/252/40222717...
4	Moi dix Mois	http://www.last.fm/music/Moi+dix+Mois	http://userserve-ak.last.fm/serve/252/54697835...
5	Bella Morte	http://www.last.fm/music/Bella+Morte	http://userserve-ak.last.fm/serve/252/14789013...

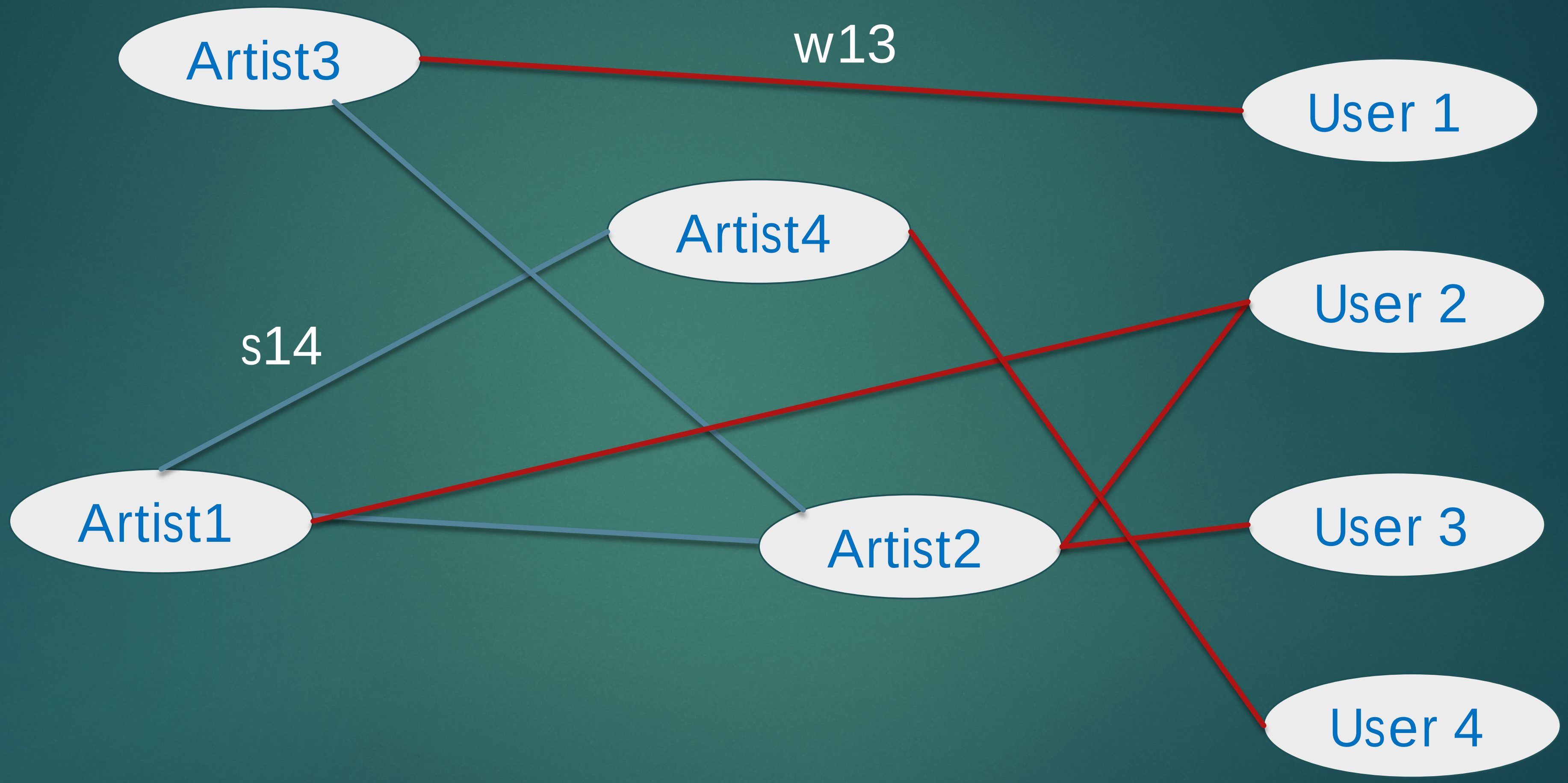
id	name	url	pictureURL	mbid
1	MALICE MIZER	http://www.last.fm/music/MALICE+MIZER	http://userserve-ak.last.fm/serve/252/10808.jpg	3897cf7f-9aac-4eef-aacb-ca0accdee9a2
2	Diary of Dreams	http://www.last.fm/music/Diary+of+Dreams	http://userserve-ak.last.fm/serve/252/3052066.jpg	22fa6038-d14c-4aab-a057-d397132e9191
3	Carpathian Forest	http://www.last.fm/music/Carpathian+Forest	http://userserve-ak.last.fm/serve/252/40222717...	69fa5c49-12ec-4c86-a238-4e07cc6d1a7d
4	Moi dix Mois	http://www.last.fm/music/Moi+dix+Mois	http://userserve-ak.last.fm/serve/252/54697835...	935d146a-ab8a-4745-b4c4-6043d4cc15c4
5	Bella Morte	http://www.last.fm/music/Bella+Morte	http://userserve-ak.last.fm/serve/252/14789013...	686dffc5-e13c-4b14-b2b7-097cbef040f0

Artist Similarity Network

- Use of mbid codes to get similar artists.
- "Match" corresponds to weight of similarity.
- Top 10 similar artists are extracted.

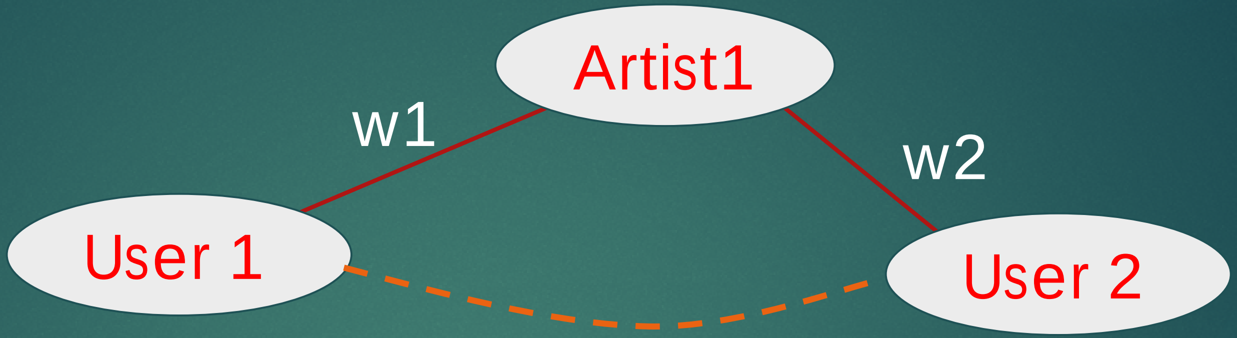
	ArtistID	Artist	Similar_artists	Weight
0	1	MALICE MIZER	Moi dix Mois	1.000000
1	1	MALICE MIZER	Közi	0.851889
2	1	MALICE MIZER	LAREINE	0.807094
3	1	MALICE MIZER	BAISER	0.595696
4	1	MALICE MIZER	Kaya	0.585429

User – Artist Network

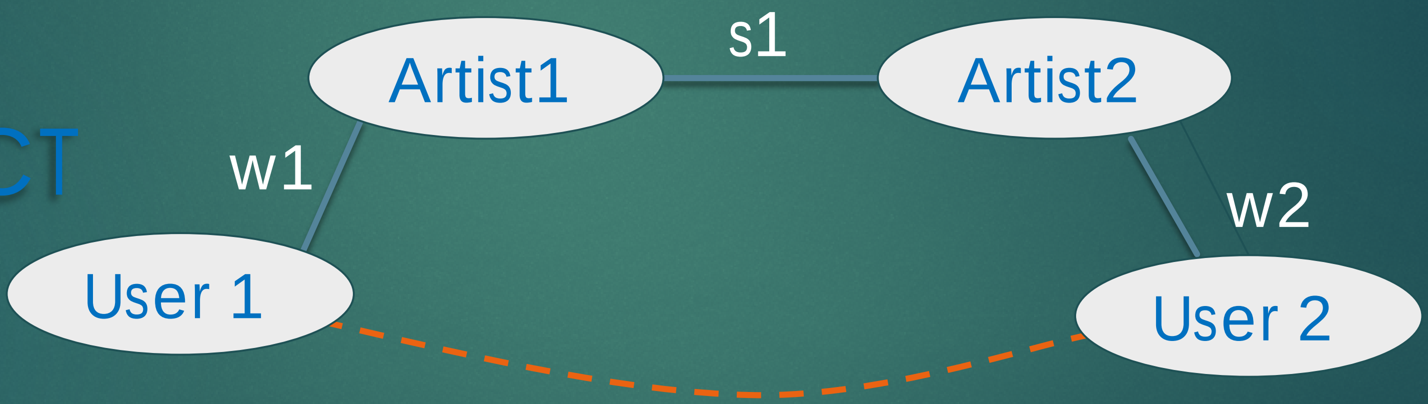


For the inference of friendships two types of connections are considered:

- DIRECT

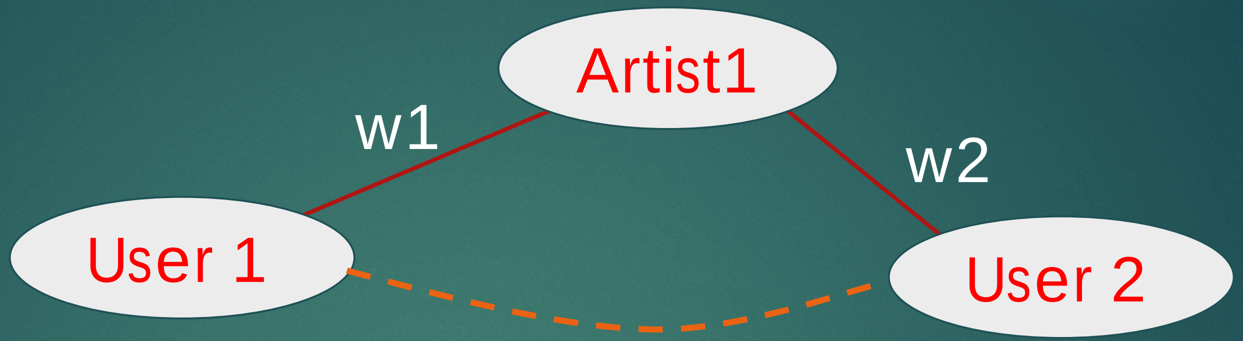


- INDIRECT

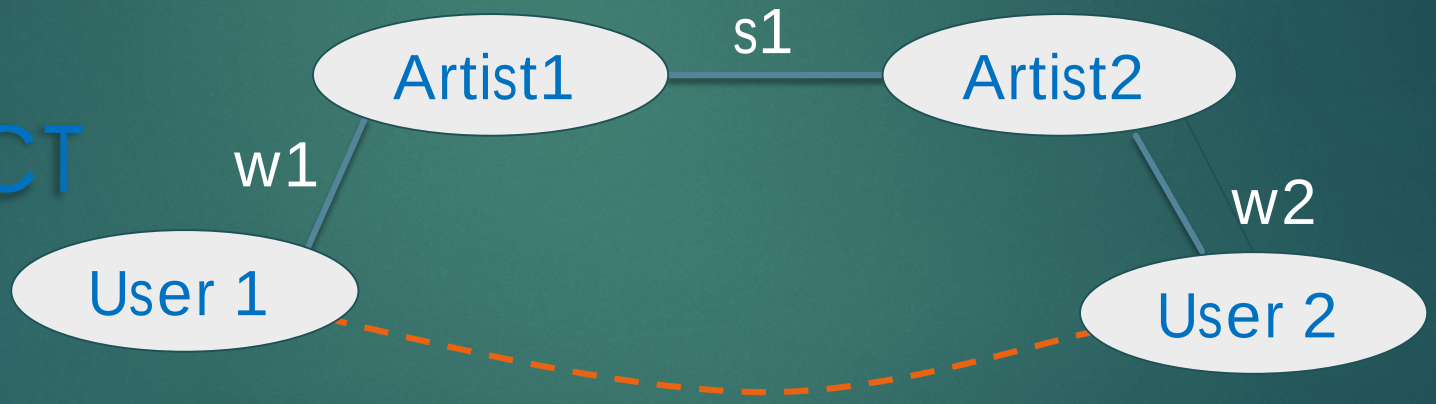


For the inference of friendships two types of connections are considered:

- DIRECT



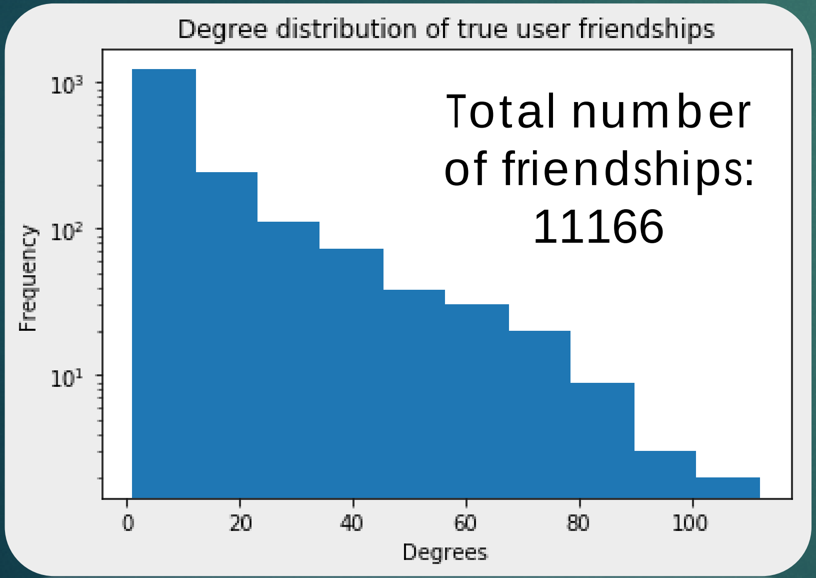
- INDIRECT



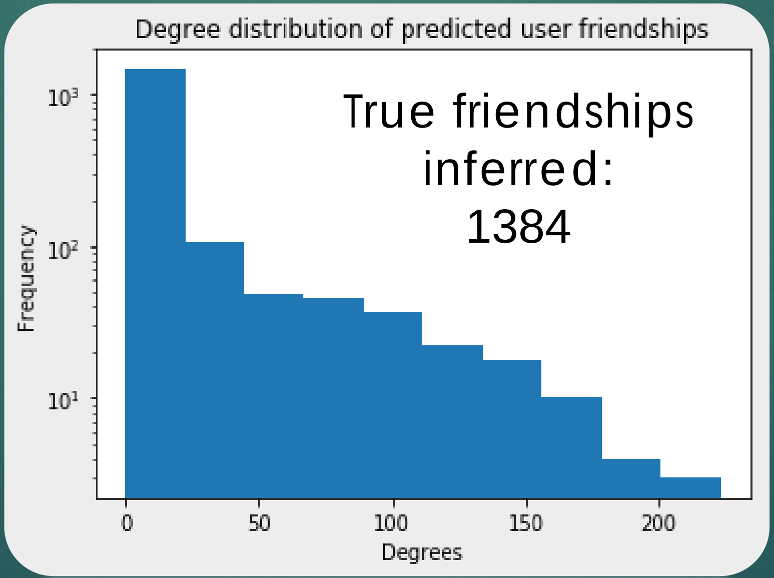
- The similarity between users *i* and *j* defined as: $F_{ij} = \sum_{k,l\in\Omega} S_{kl}\min\{C_{ik},C_{jl}\} + \sum_{k\in\Theta}\min\{C_{ik},C_{jk}\}$
- Friendship inferred if the similarity is above a *threshold.*

- Random artist similarity network is generated using the configuration model.
- Friendships between users inferred using this random network.
- Performance evaluated as the number of existing friendships detected and compared with the inferences made using the artist similarity network.

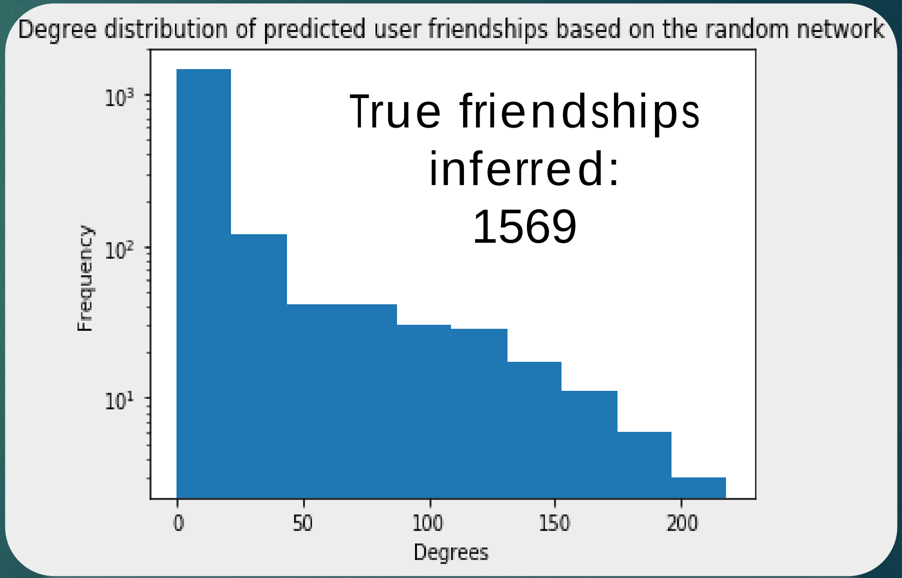
Original social network



Inferred from artist network



Inferred from random network



Recommender System

- Use of a famous and well tested algorithm as a baseline.
- Introduction of the artist similarity and social networks inside the prediction algorithm.
- Optimization of execution time.
- Comparison of the two networks.

Recommender System

- Use of a famous and well tested algorithm as a base.
- Introduction of the artist similarity and social networks inside the prediction algorithm.
- Optimization of execution time.
- Comparison of the two networks.

- SVD from the Netflix Prize
- Smooth SVD
- Cython optimizer
- Grid search with cross validation
  - Ratings rescaled from 1 to 6
  - 6 folds
  - Grid search over regularizer coefficients

Smooth SVD

$$\widehat{r_{ui}} = \mu + b_u + b_i + w_u^T z_i$$

$$F_i = ||R_{:i} - \hat{R}_{:i}||^2 + \alpha R_{:i}^T L \hat{R}_{:i}$$

$$F_{ui} = (r_{ui} - \hat{r_{ui}})^2 + \alpha \sum_{k=1}^{|U|} L_{uk} \hat{r_{ui}} \hat{r_{ki}}$$

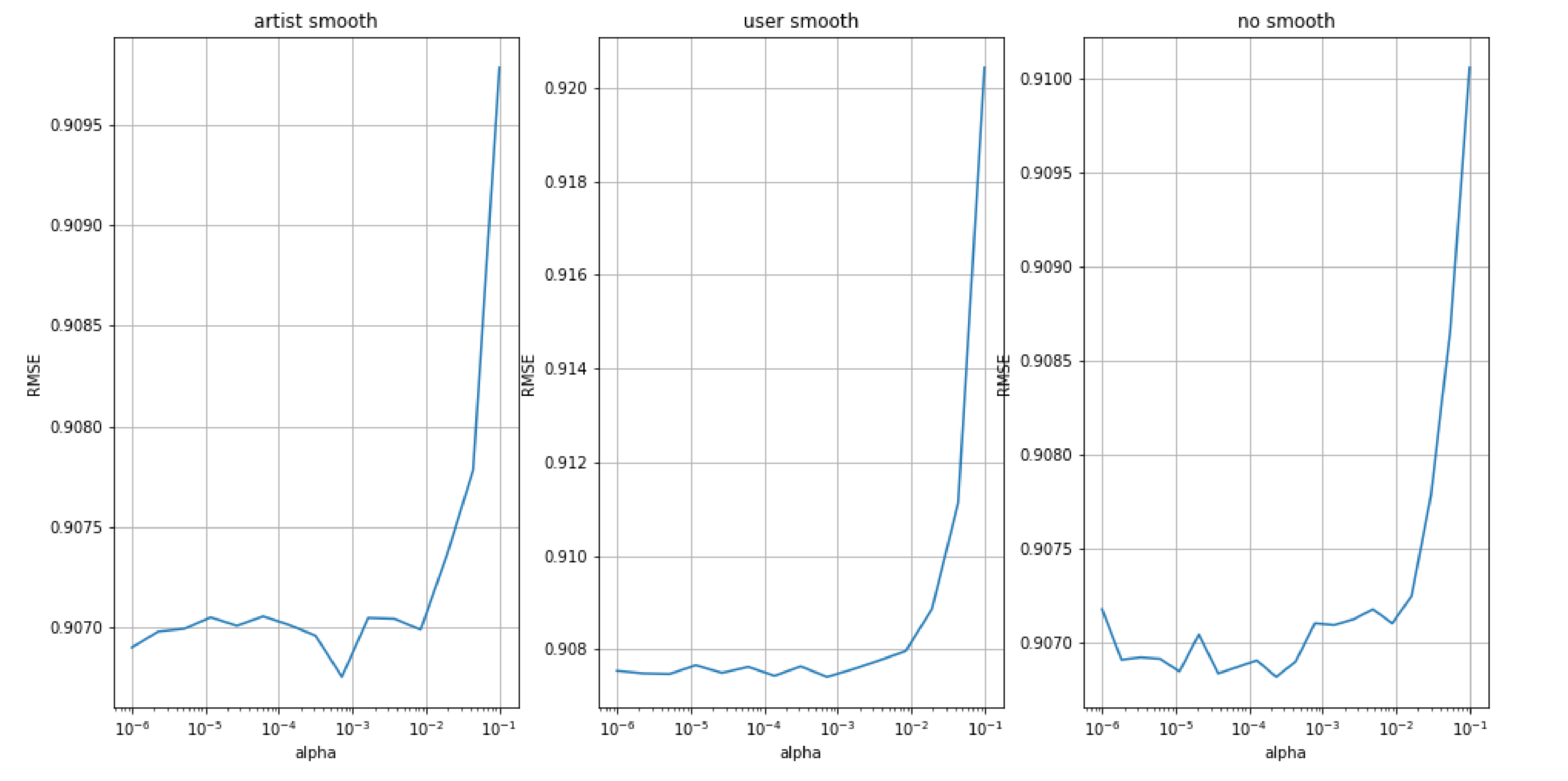
$$\frac{\partial F_{ui}}{\partial b_u} = -2(r_{ui} - \hat{r_{ui}}) + \alpha \sum_{k=1}^{|U|} L_{uk}(\hat{r_{ki}} + \hat{r_{ui}})$$

$$\frac{\partial F_{ui}}{\partial b_i} = -2(r_{ui} - \hat{r_{ui}}) + \alpha \sum_{k=1}^{|U|} L_{uk}(\hat{r_{ki}} + \hat{r_{ui}})$$

$$\frac{\partial F_{ui}}{\partial w_u} = -2(r_{ui} - \hat{r_{ui}}) z_i + \alpha z_i \sum_{k=1}^{|U|} L_{uk}(\hat{r_{ki}} + \hat{r_{ui}})$$

$$\frac{\partial F_{ui}}{\partial z_i} = -2(r_{ui} - \hat{r_{ui}}) w_u + \alpha \sum_{k=1}^{|U|} L_{uk}(w_u \hat{r_{ki}} + w_k \hat{r_{ui}})$$

RMSE on grid search



Interpretation

- Real ratings are not smooth over neither of the two networks
- Smoothing does not help with this particular algorithm
- Possible bug in the smooth SVD implementation