

Steam Explorer: Games Recommendation for Steam Users

Jiajia Liu, Jia Tan, Yixin Peng



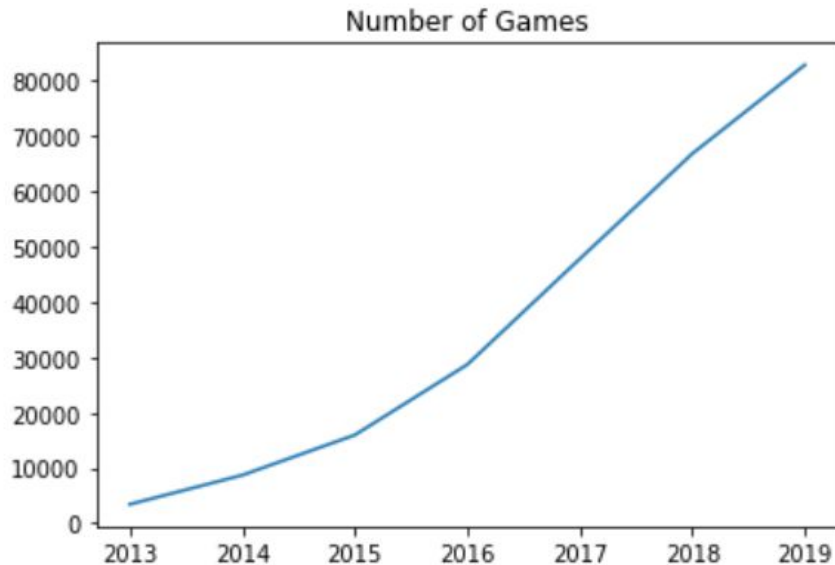
What's Steam?

An internet-based video game digital distribution, digital rights management, multiplayer and social networking program. [\[1\]](#)



Why Steam Games Recommendation?

Allow Steam users easily find games.

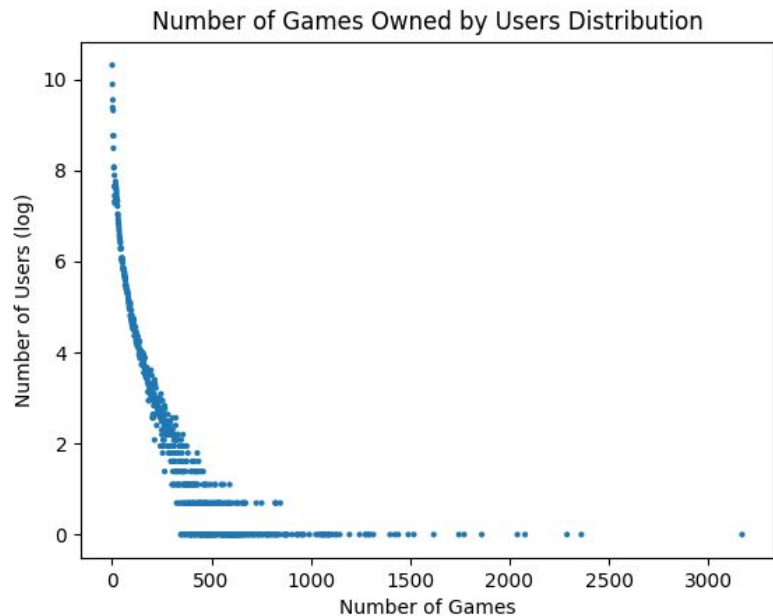


Overview

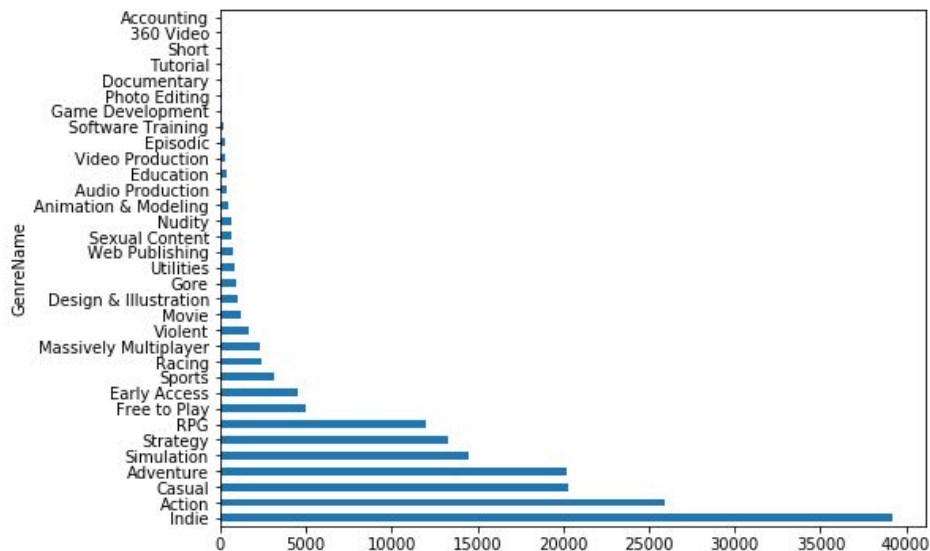
- Data
- Approach
- Evaluation
- Results
- App

Data: Collection

User-game pairs collected by [BYU \[2\]](#) in 2014.
(85,681,026 pairs, 178,454 users, 4,466 games)



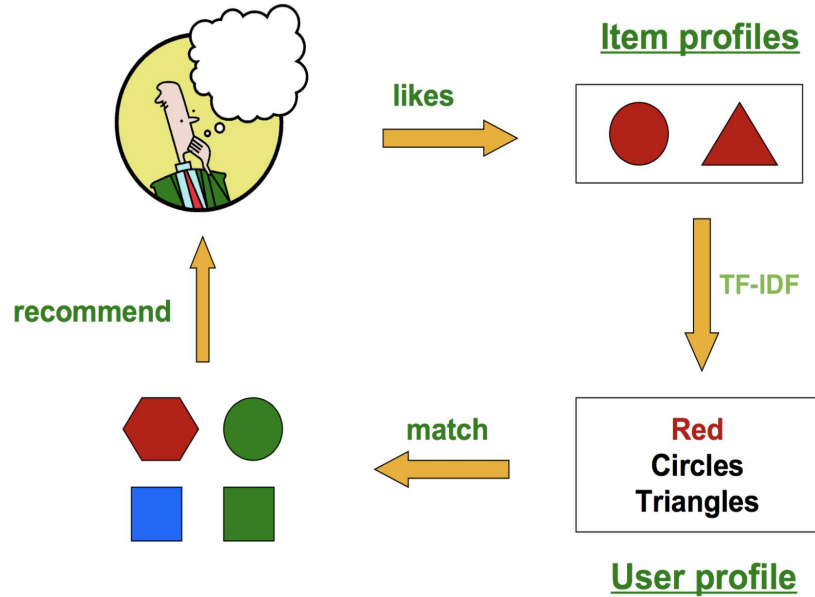
Games features collected using [Steam API \[3\]](#).



Data: Preprocess

- Select user-games-pairs with users' game number between 10 to 200 and delete games which playtime is 0. (1,319,053 pairs, 38,308 users, 4,192 games)
- Game description lemmatization.
- Split 80% of data for model training and 20% for testing. (K-Fold, K=5)

Approach: Content-based



- Tokenize description of each game --> features
- Build profile of each User-Game-pair
- Rank games for each user, and recommend top-N games to user
- Try to add other features(genre, price)
- Optimization parameter: "min_df" "max_df" and "isOtherFeatures"

Approach: User-User Collaborative Filtering

	1	2		n-1	
1					
2					
\uparrow	R	-		R	
S_{ij}					
\downarrow	R	R		R	-
m-1					

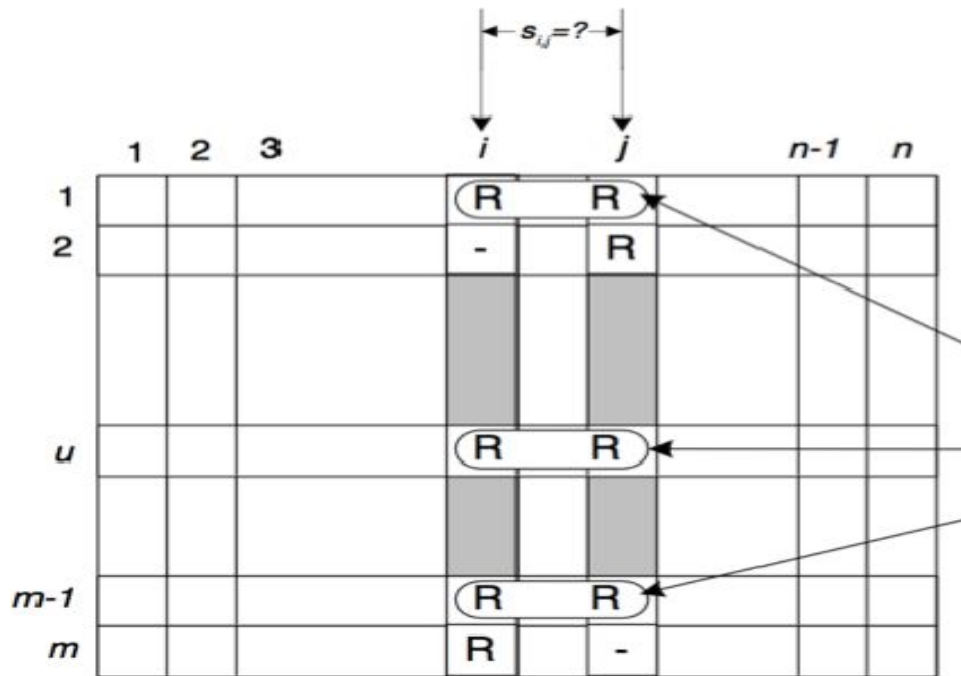
- According to each user's playtime to score games
- Use cosine similarity measure to find user-user similarity

$$\text{sim}(\mathbf{x}, \mathbf{y}) = \cos(\mathbf{r}_x, \mathbf{r}_y) = \frac{\mathbf{r}_x \cdot \mathbf{r}_y}{\|\mathbf{r}_x\| \cdot \|\mathbf{r}_y\|}$$

- Using similar user's preference to recommend another user

$$r_{xi} = \frac{\sum_{y \in N} s_{xy} \cdot r_{yi}}{\sum_{y \in N} s_{xy}}$$

Approach: Item-Item Collaborative Filtering



- According to each user's playtime to score games
- Use cosine similarity measure to find item-item similarity
- Estimate rating for one game based on ratings for similar games and the prediction function is same as user based model

$$r_{xi} = \frac{\sum_{j \in N(i;x)} s_{ij} \cdot r_{xj}}{\sum_{j \in N(i;x)} s_{ij}}$$

Evaluation Metrics

Recall Rate:

- We will recommend top-N games
- If the game's rating is greater than top-N's min rating, then hit+1
- Finally, the number of all hit games/the number of games really owned by the users is the recall rate

$$R = \frac{TP}{TP+FN}$$

RMSE:

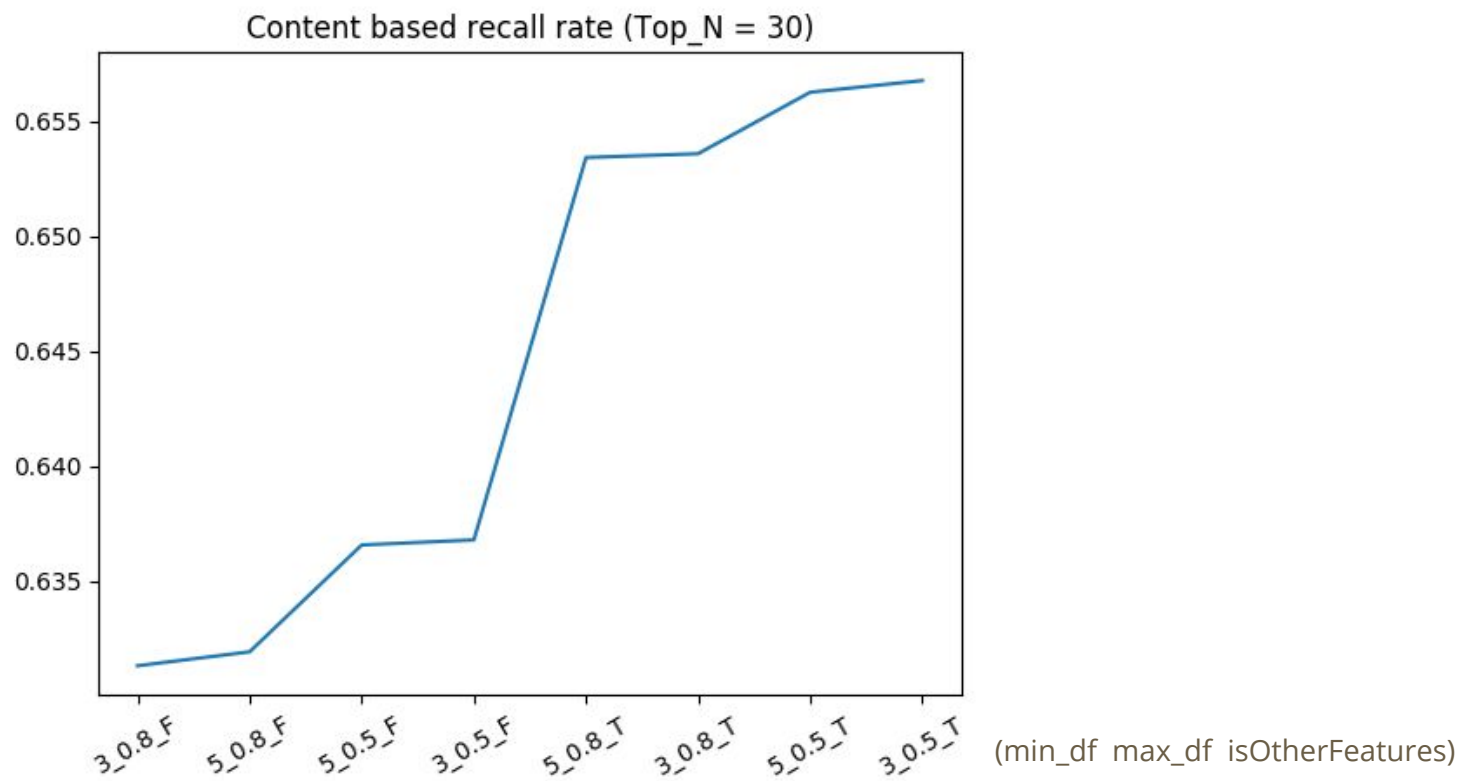
- Get the predict matrix and test matrix
- Use these two matrix to calculate the RMSE value for the recommender model
- Compare all models' RMSE, and the minimum one is the best

Root-mean-square error (RMSE)

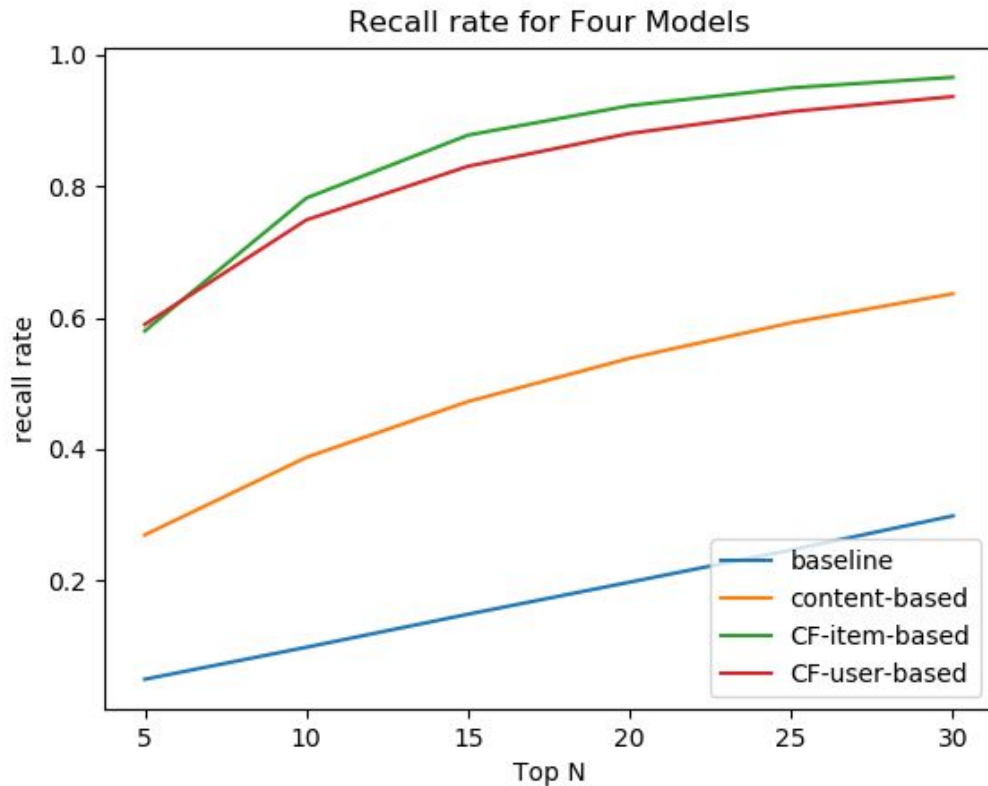
▪ $\sqrt{\sum_{xi} (r_{xi} - r_{xi}^*)^2}$ where r_{xi} is predicted, r_{xi}^* is the true rating of x on i

Results: Recall

Content based:



Results: Recall

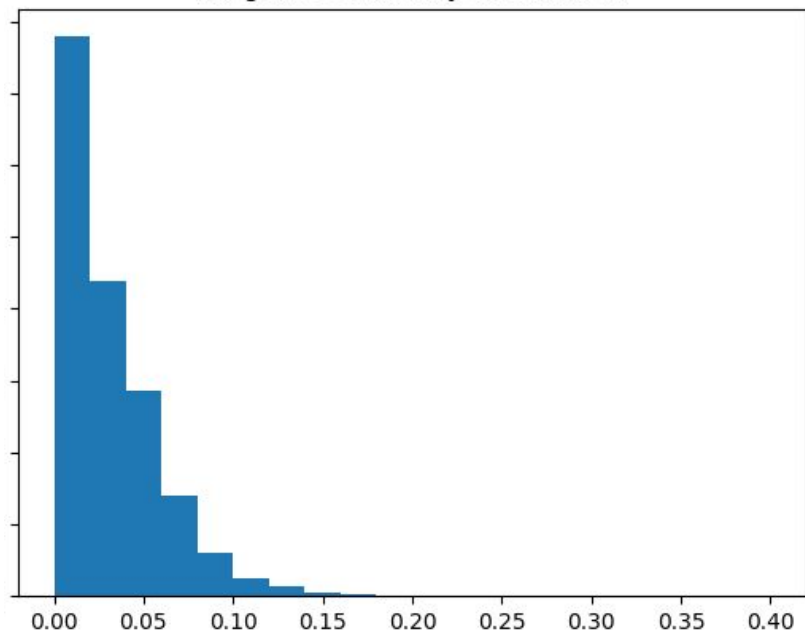


*baseline:

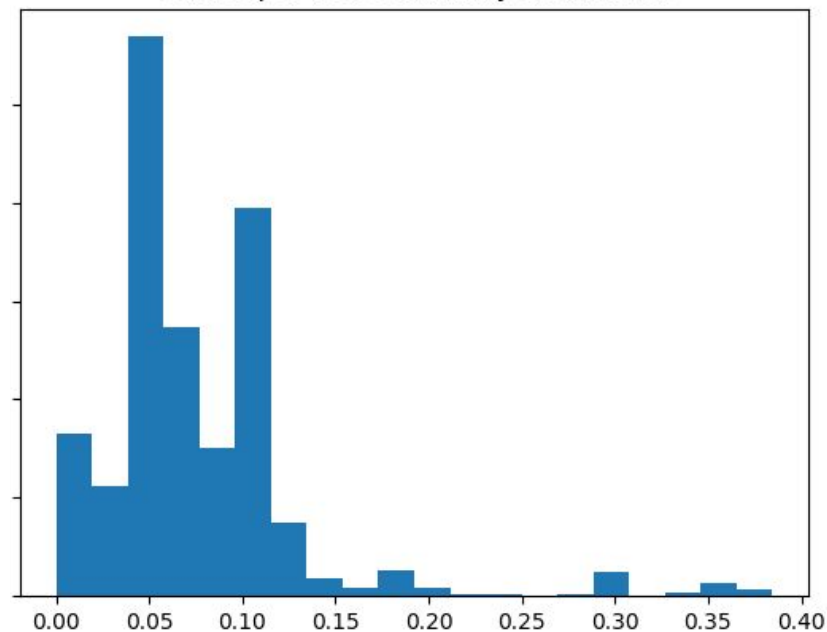
Random rating with
gaussian distribution

Results: Content Similarity

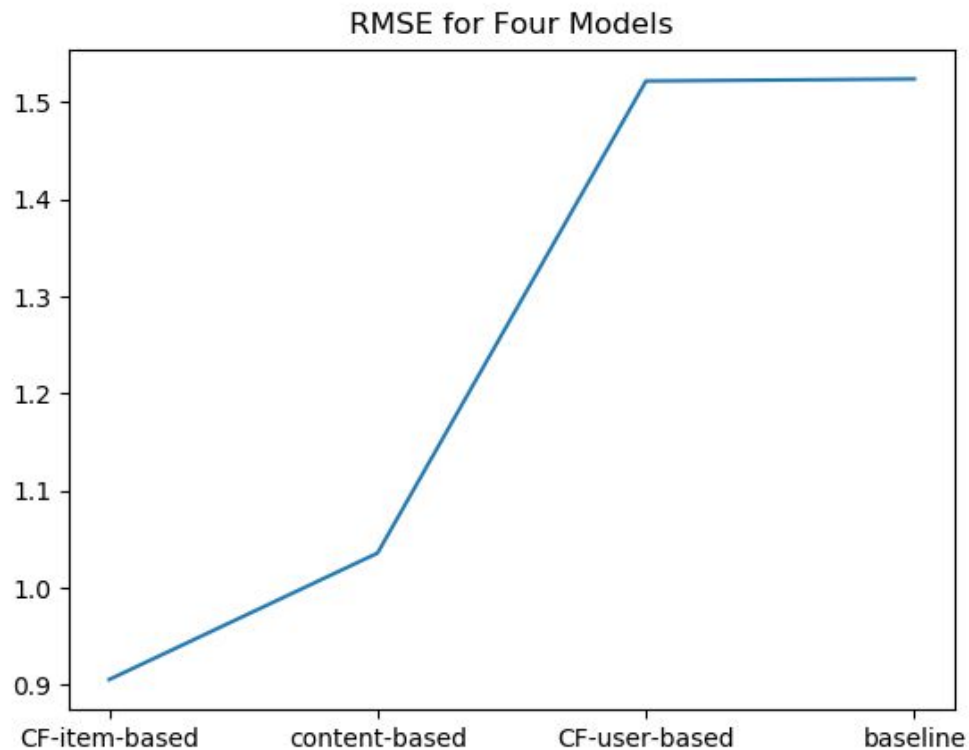
All games similarity distribution



Games per users similarity distribution



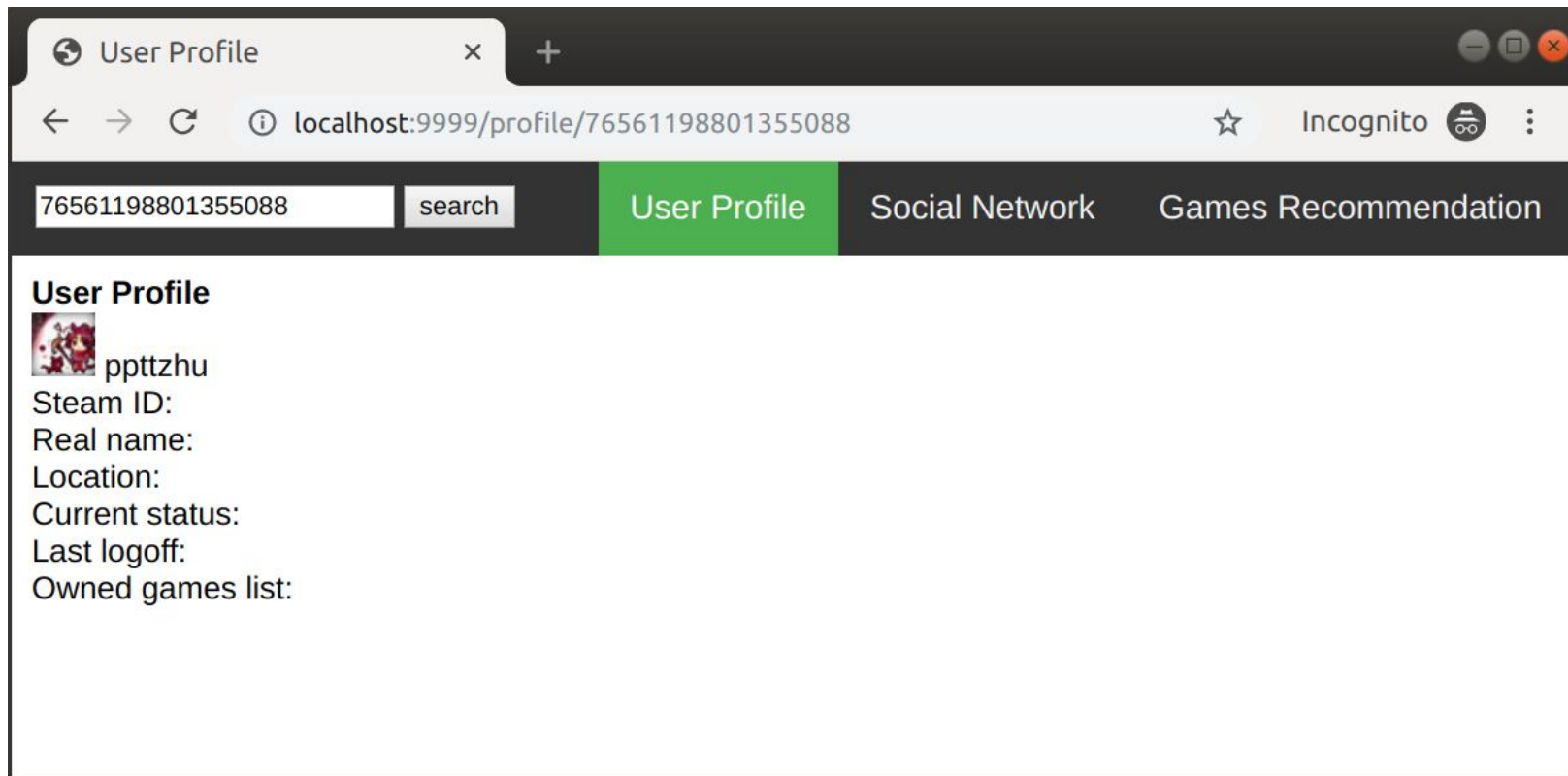
Results: RMSE



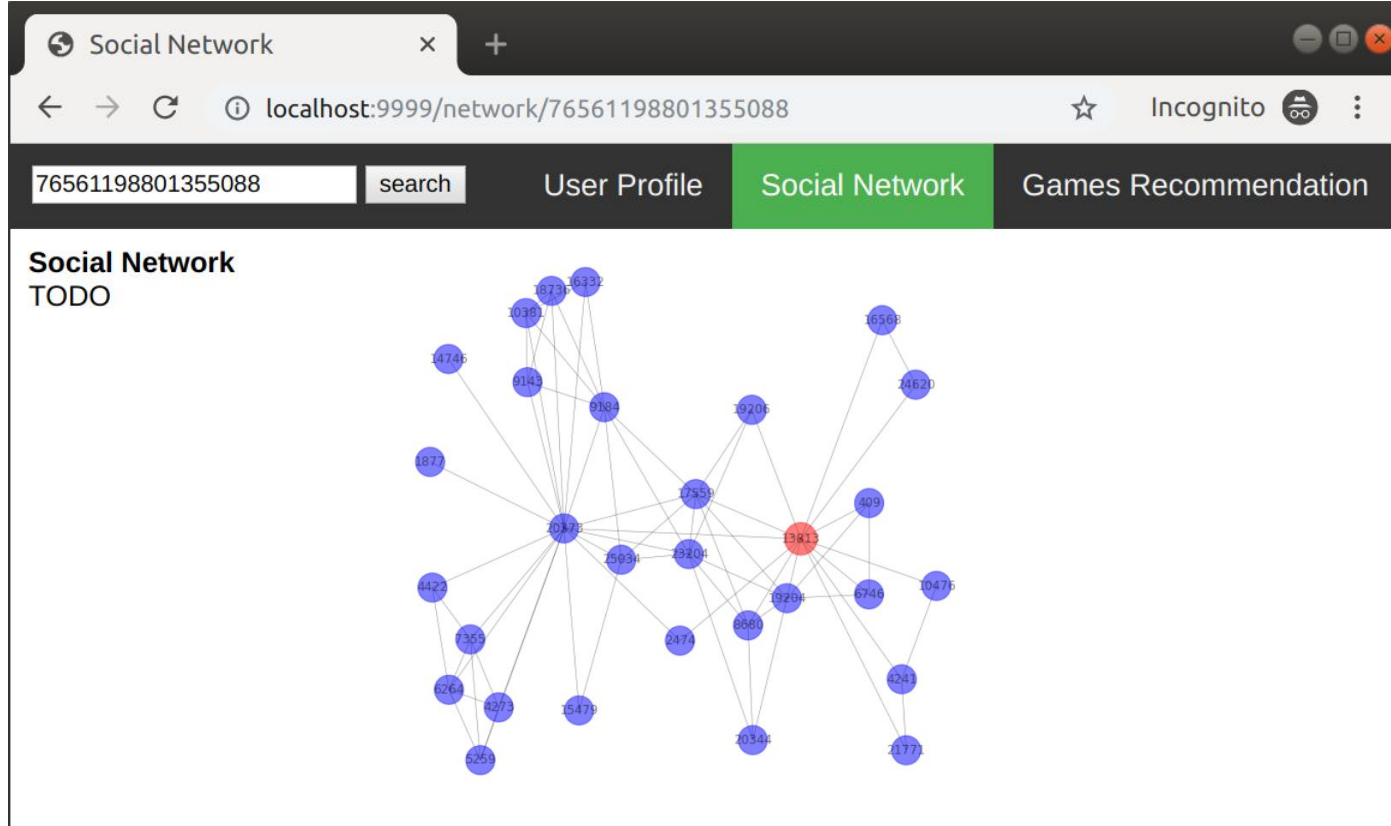
*baseline:

Random rating with
gaussian distribution

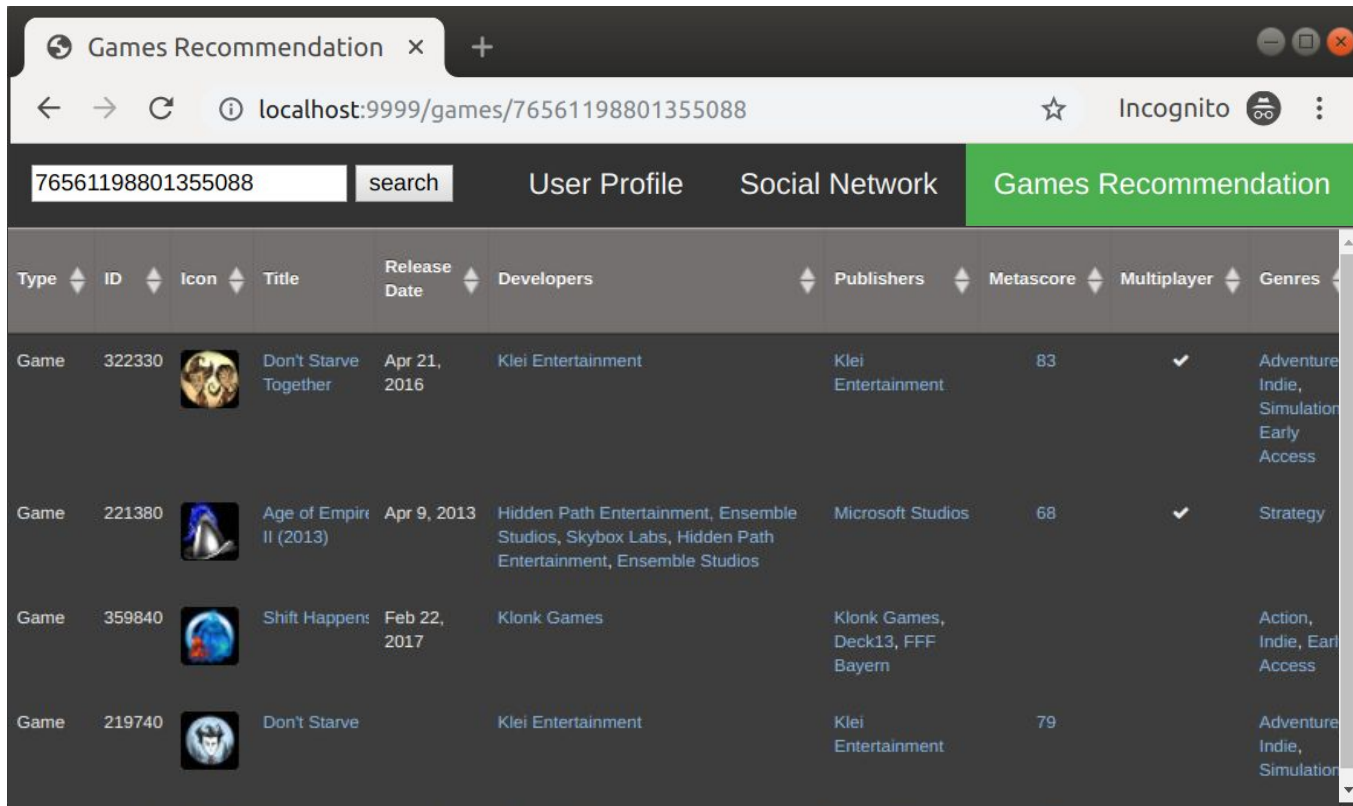
App (demo)







App (demo)



App (demo)



76561198801355088 search User Profile Social Network Games Recommendation

Type	ID	Icon	Title	Release Date	Developers	Publishers	Metascore	Multiplayer	Genres
Game	322330		Don't Starve Together	Apr 21, 2016	Klei Entertainment	Klei Entertainment	83	✓	Adventure, Indie, Simulation, Early Access
Game	221380		Age of Empires II (2013)	Apr 9, 2013	Hidden Path Entertainment, Ensemble Studios, Skybox Labs, Hidden Path Entertainment, Ensemble Studios	Microsoft Studios	68	✓	Strategy
Game	359840		Shift Happens	Feb 22, 2017	Klonk Games	Klonk Games, Deck13, FFF Bayern			Action, Indie, Early Access
Game	219740		Don't Starve		Klei Entertainment	Klei Entertainment	79		Adventure, Indie, Simulation

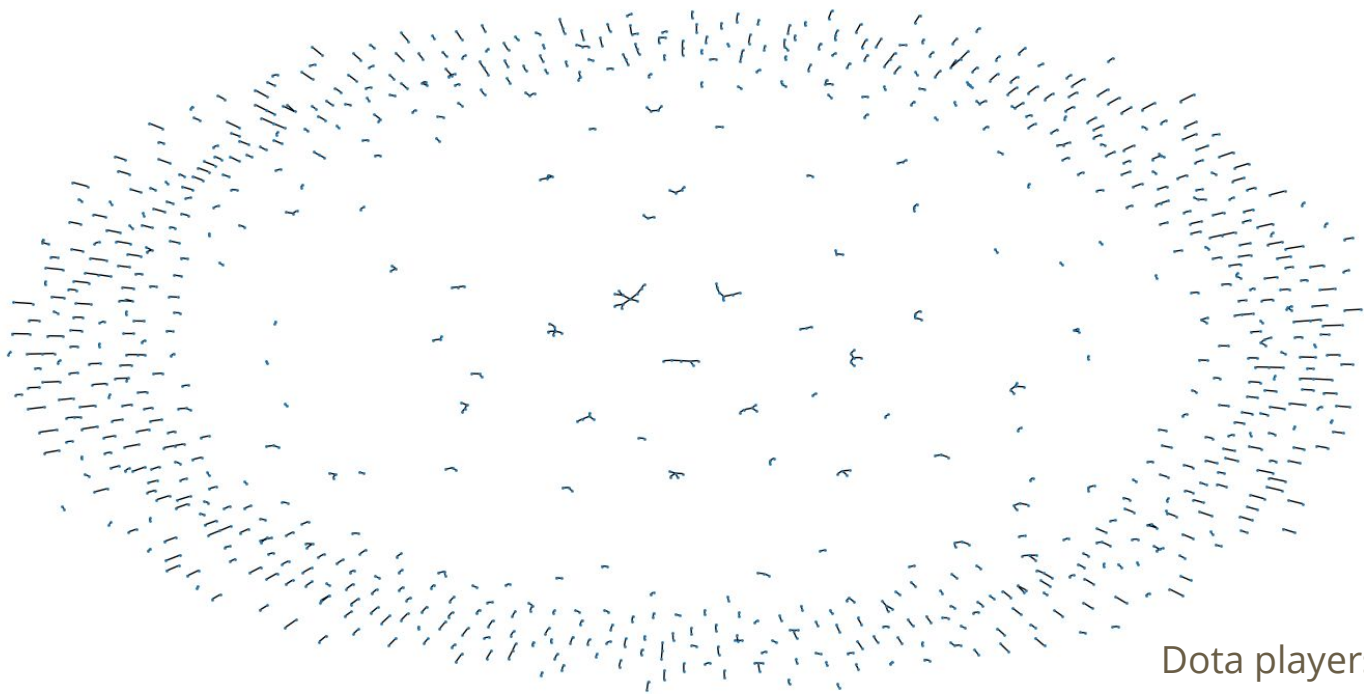
Thank you!



Questions?

Appendix: Friends Recommendation

Performance is poor because steam does not have a strong social network.



Dota players in 2014

Appendix: Reference

- [1] Steam (software) from wikipedia, [https://simple.wikipedia.org/wiki/Steam \(software\)](https://simple.wikipedia.org/wiki/Steam_(software))
- [2] Historical data collected by Brigham Young University, <https://steam.internet.byu.edu/>
- [3] Live data collected using Steam API, [https://developer.valvesoftware.com/wiki/Steam Web API](https://developer.valvesoftware.com/wiki/Steam_Web_API)

Appendix: Contributions

Tasks	Jiajia Liu	Jia Tan	Yixin Peng
Data collection	Crawled real-time games data using API.	Crawled real-time User-Friends-Pair using API.	Crawled real-time User-Games-Pair using API.
Data pre-processing	Pre-processed games description text and game features.	Perform statistics analysis.	Select subset from historical dataset. Stem Vocabularies.
Model training	Add game features to content-based game recommendation.	Trained friends link prediction model.	Trained content-based game recommendation model.
Model evaluation	Evaluated collaborative filtering model.	Evaluated friends link prediction model. Calculated RMSE evaluation.	Used recall rate to evaluated content-based game recommendation.