Project - Recommender Systems

Group 3

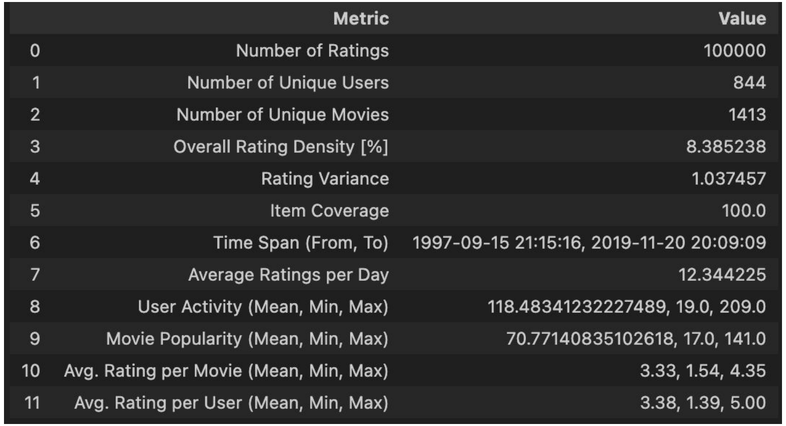
Content

- **Data set & Data pre-processing**
- **Individual Recommender**
  - Content
  - Evaluation
  - Explanation
  - Evaluation of explanations + Results
- **Group Recommender**
  - Content
  - Evaluation
  - Explanation
  - Evaluation of explanations + Results

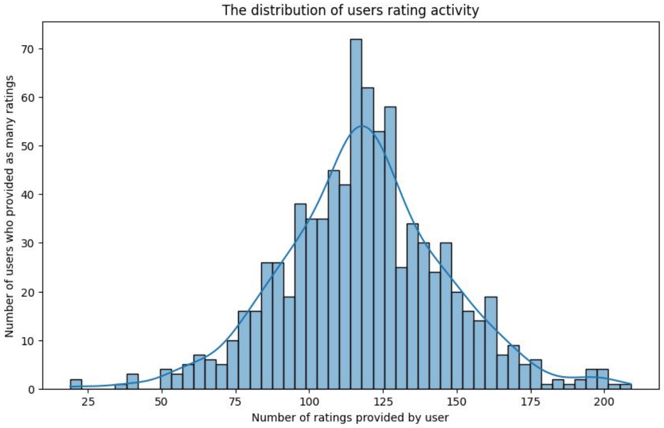
Data set and Data Pre-processing

- **Chosen dataset** - MovieLens 25M (movies.csv, ratings.csv) - Select 100k subset of it.
- We analyze the distribution of the subset obtained to make an informed decision on the data.
- We obtain 0.26% rating density (quite sparse), therefore we select a less-sparse subset according to the following steps:
    1. **Filtering users**: selecting users who have rated at least 10*mean_user_activity items. This helps ensure that users have a high level of engagement with the dataset.
    2. **Filtering items**: selecting movies that have received a minimum number of 10*mean_item_popularity ratings. This ensures that items in the subset are relatively popular.
    3. **Random Sampling**: random sampling to select 100,000 ratings from the filtered dataset.

New 100k obtained from the filtered subset

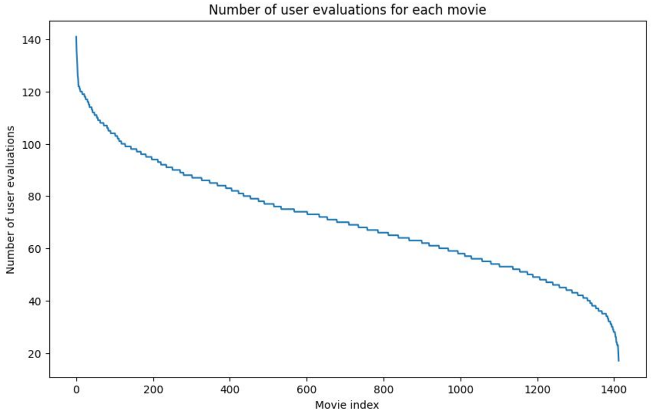


	Metric	Value
0	Number of Ratings	100000
1	Number of Unique Users	844
2	Number of Unique Movies	1413
3	Overall Rating Density [%]	8.385238
4	Rating Variance	1.037457
5	Item Coverage	100.0
6	Time Span (From, To)	1997-09-15 21:15:16, 2019-11-20 20:09:09
7	Average Ratings per Day	12.344225
8	User Activity (Mean, Min, Max)	118.48341232227489, 19.0, 209.0
9	Movie Popularity (Mean, Min, Max)	70.77140835102618, 17.0, 141.0
10	Avg. Rating per Movie (Mean, Min, Max)	3.33, 1.54, 4.35
11	Avg. Rating per User (Mean, Min, Max)	3.38, 1.39, 5.00



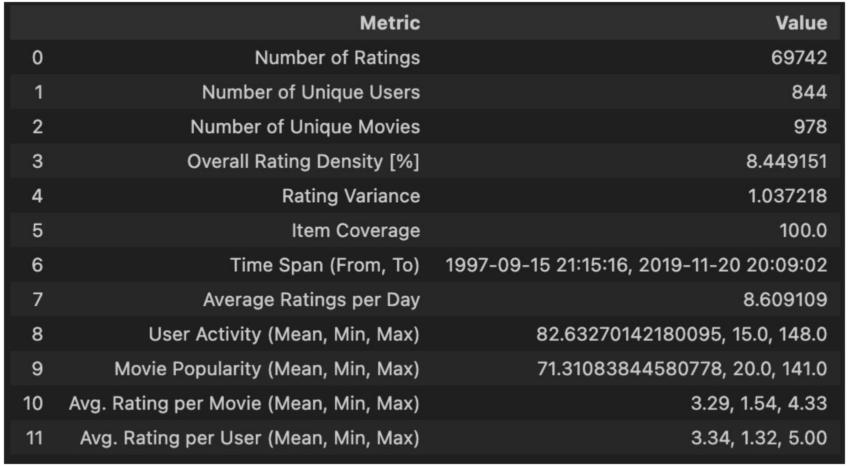
- As we can see, with this preprocessing steps we were able to filter a 100K subset of MovieLens 25M that has a higher density (8.4%) than the MovieLens 100K dataset (6.3%) (movielens-small) which is described in Harper, et al. (2015) "The MovieLens Datasets: History and Context".

- Also, we got rid of the "long tail" distribution! Now the first graph resembles the Gaussian distribution! This is a very satisfactory result, thus after some final following preprocessing steps we can save it and use it for our recommenders. + We remove duplicates.

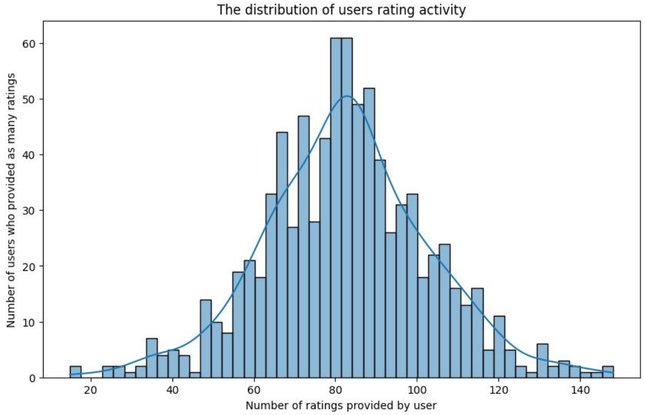


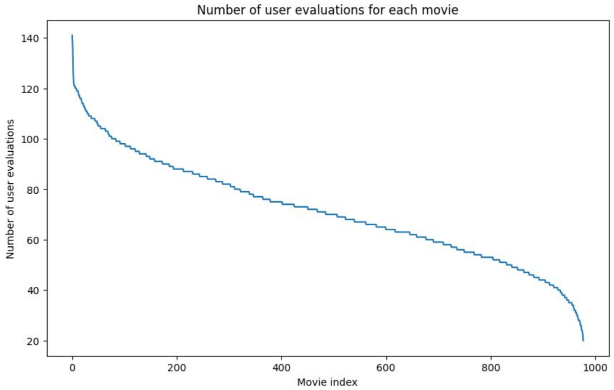
Merging dataset with Movie metadata - Wikipedia Movie Plots

- We obtain title, year, cast, director, origin, plot.
- We filter out movies with a considerable lack of information from this new dataset.
- We merge both datasets, and re-evaluate the data statistics:



	Metric	Value
0	Number of Ratings	69742
1	Number of Unique Users	844
2	Number of Unique Movies	978
3	Overall Rating Density [%]	8.449151
4	Rating Variance	1.037218
5	Item Coverage	100.0
6	Time Span (From, To)	1997-09-15 21:15:16, 2019-11-20 20:09:02
7	Average Ratings per Day	8.609109
8	User Activity (Mean, Min, Max)	82.63270142180095, 15.0, 148.0
9	Movie Popularity (Mean, Min, Max)	71.31083844580778, 20.0, 141.0
10	Avg. Rating per Movie (Mean, Min, Max)	3.29, 1.54, 4.33
11	Avg. Rating per User (Mean, Min, Max)	3.34, 1.32, 5.00

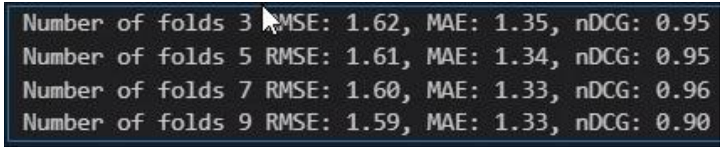




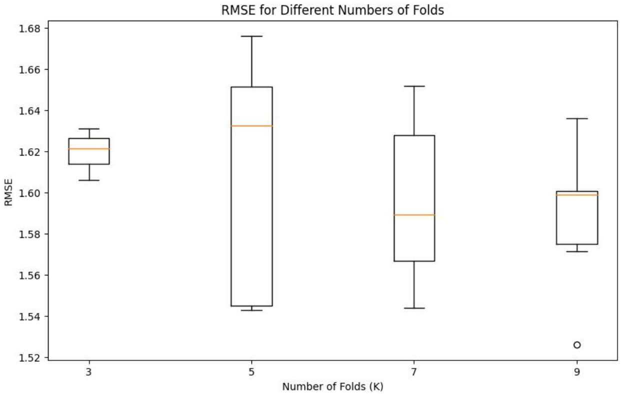
Individual Recommender - Baselines

Decision: Why RMSE as a main metric? We do not want the users to be unsatisfied. Therefore we aim to minimize it. Gives bigger weight to outliers compared to other metrics.

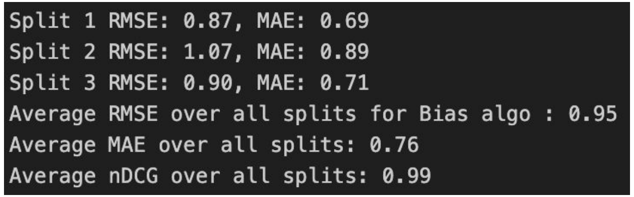
- First, we implement basic.Popular. Pros and cons are described in the Appendix.
- Data is split into test and train using K-fold cross validation.
  - This has the advantage of evaluating the algorithm on different subsets of the data each time so we can have a better conclusion how it performs on different unseen data sets.
  - We experimented using different values for the number of folds (3,5,7,9) to see which one will obtain the lowest root mean squared error.
  - 3-fold has the smallest deviation - more reliable



```
Number of folds 3 RMSE: 1.62, MAE: 1.35, nDCG: 0.95
Number of folds 5 RMSE: 1.61, MAE: 1.34, nDCG: 0.95
Number of folds 7 RMSE: 1.60, MAE: 1.33, nDCG: 0.96
Number of folds 9 RMSE: 1.59, MAE: 1.33, nDCG: 0.90
```



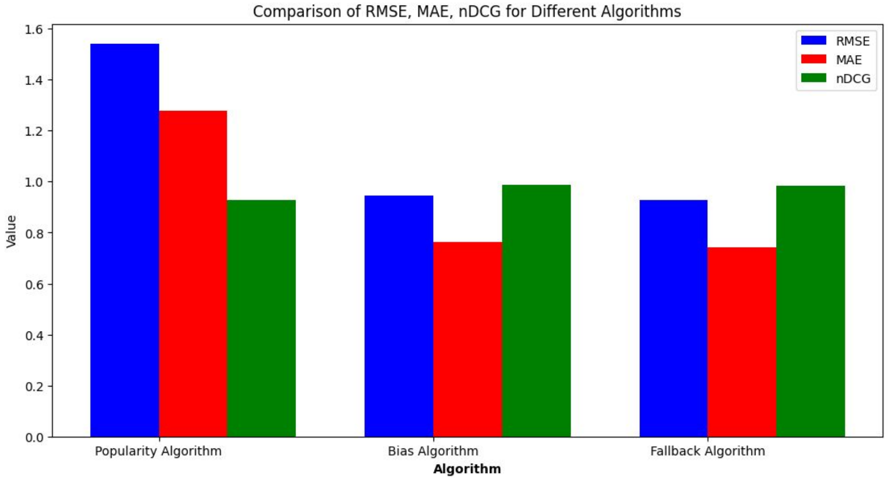
Individual Recommender - Baselines

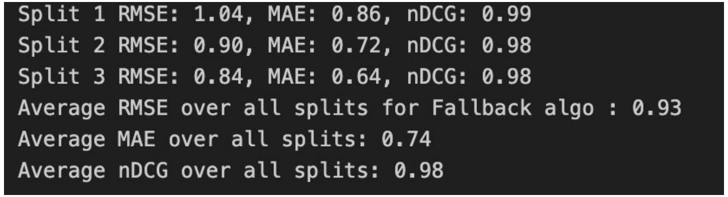


```
Split 1 RMSE: 0.87, MAE: 0.69
Split 2 RMSE: 1.07, MAE: 0.89
Split 3 RMSE: 0.90, MAE: 0.71
Average RMSE over all splits for Bias algo : 0.95
Average MAE over all splits: 0.76
Average nDCG over all splits: 0.99
```

- Second baseline implemented is Bias recommender.
- Same experiment is ran. (More about it in Appendix)

- We have a third recommender. We combine the two previous baselines and use the fallback algorithm which will take as parameters both the popularity and bias algorithms. We also use k-fold validation with 3 splits.





```
Split 1 RMSE: 1.04, MAE: 0.86, nDCG: 0.99
Split 2 RMSE: 0.90, MAE: 0.72, nDCG: 0.98
Split 3 RMSE: 0.84, MAE: 0.64, nDCG: 0.98
Average RMSE over all splits for Fallback algo : 0.93
Average MAE over all splits: 0.74
Average nDCG over all splits: 0.98
```

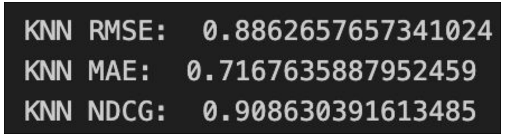
- We will use Bias as a baseline comparator.

Individual Recommender - Metrics

- **Why do we use RMSE and MAE?**
  - Lenskit provides both of them, easy to implement and understand.
  - Both evaluation methods are sensible to data sampling, what we also did with our large dataset.
  - MAE gives as clear interpretation of its score (example MAE = 0.7 means that on average our predictions are off for 0.7 rating score)
  - RMSE is more sensible to outliers so it will help to better recommendations, not having ones which are completely off.
  - nDCG to compare how good our recommendations are, takes rank into account.

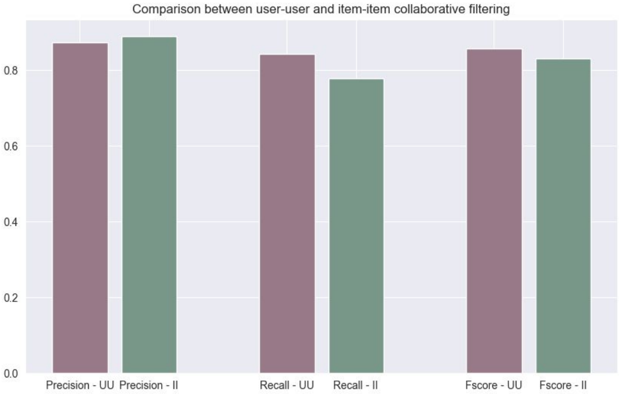
Individual Recommender - Advanced

- Content based and collaborative filtering (user-based and item-based).
- We start by pre-processing data for content based/matrix factorization.
  - We will use plot, title and genre for our TF-IDF vector. (These 3 have the highest important by features)
- **Content-based:**
  - KNN - cosine metric, 12 neighbours.
  - We calculate RMSE, MAE and nDCG.
  - We use hold-out method (80-20)



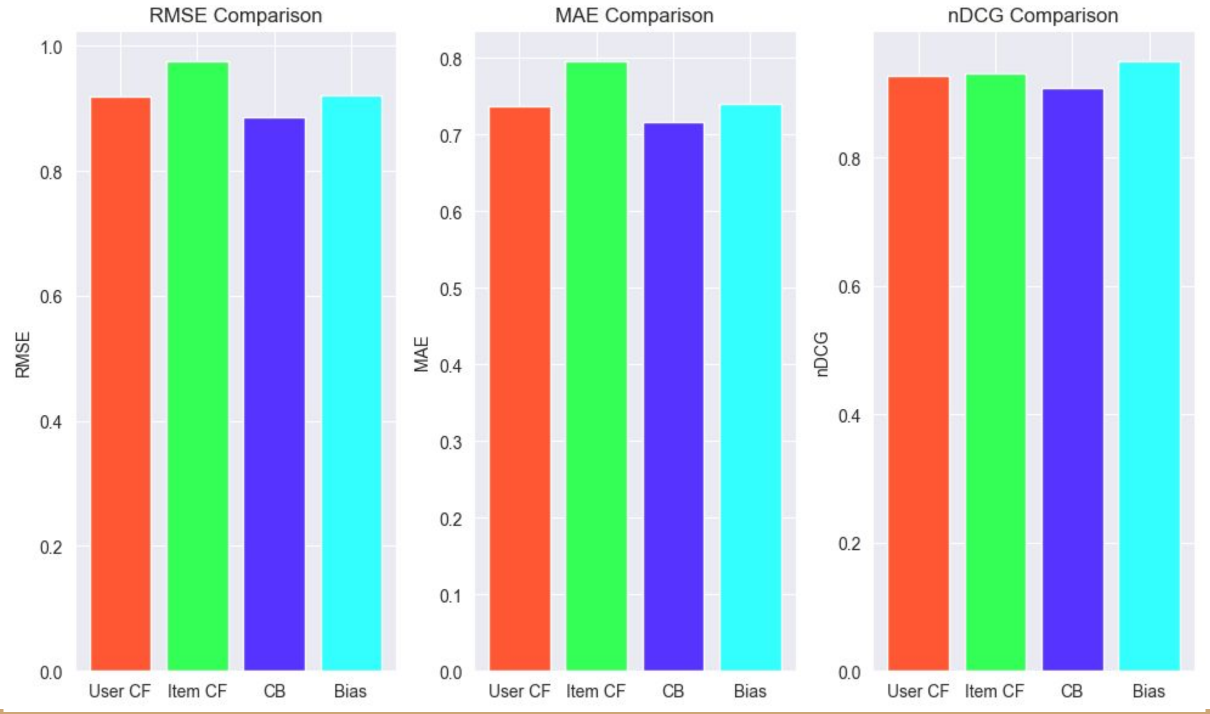
```
KNN RMSE:  0.8862657657341024
KNN MAE:  0.7167635887952459
KNN NDCG:  0.908630391613485
```

- **Collaborative filtering:** (Aim to prove the hypothesis shown in class - there is no significant difference in results considering user-based and item-based CF)
  - Hold-out method to compare user-user and item-tem CF - We will use precision, recall and F1 Score.
  - We expect them to perform the same as discussed in class.



Evaluation IR - Results

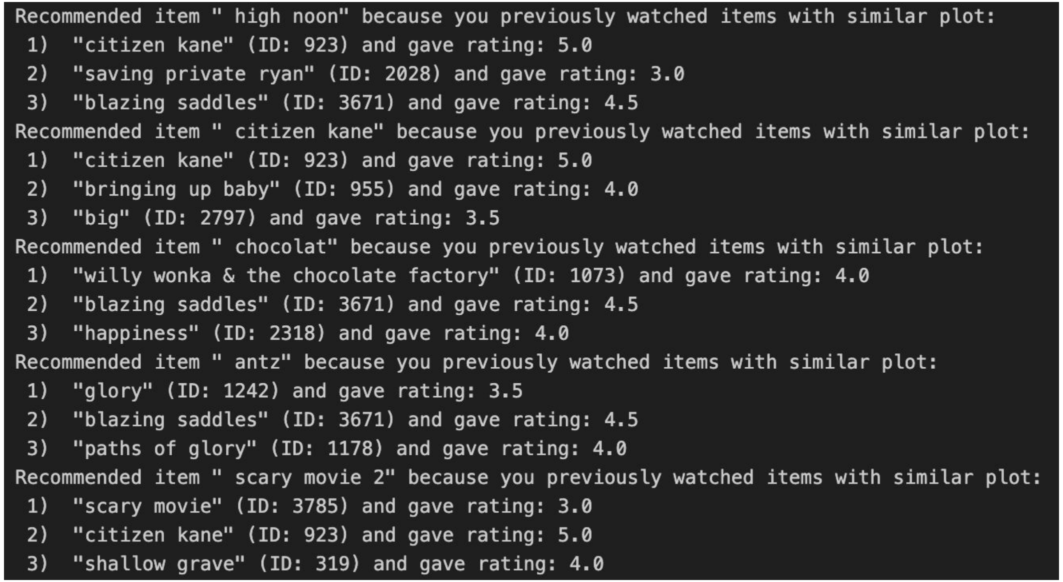
- We once again calculate RMSE, MAE and nDCG with K-fold for CFs and compare with the other algorithms.



Explanations - Individual Recommender

Decision: Using content-based to get recommendations.

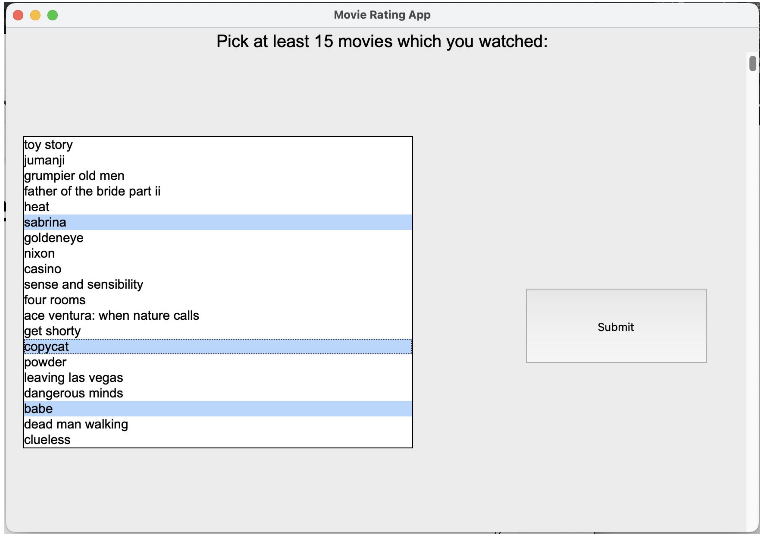
- We use content-based to get recommendations as it performs the best from other individual recommenders.
- Template-based explanations - Recommended item ‘item_title’ because you previously watched items with similar plot.



```
Recommended item " high noon" because you previously watched items with similar plot:
 1)  "citizen kane" (ID: 923) and gave rating: 5.0
 2)  "saving private ryan" (ID: 2028) and gave rating: 3.0
 3)  "blazing saddles" (ID: 3671) and gave rating: 4.5
Recommended item " citizen kane" because you previously watched items with similar plot:
 1)  "citizen kane" (ID: 923) and gave rating: 5.0
 2)  "bringing up baby" (ID: 955) and gave rating: 4.0
 3)  "big" (ID: 2797) and gave rating: 3.5
Recommended item " chocolat" because you previously watched items with similar plot:
 1)  "willy wonka & the chocolate factory" (ID: 1073) and gave rating: 4.0
 2)  "blazing saddles" (ID: 3671) and gave rating: 4.5
 3)  "happiness" (ID: 2318) and gave rating: 4.0
Recommended item " antz" because you previously watched items with similar plot:
 1)  "glory" (ID: 1242) and gave rating: 3.5
 2)  "blazing saddles" (ID: 3671) and gave rating: 4.5
 3)  "paths of glory" (ID: 1178) and gave rating: 4.0
Recommended item " scary movie 2" because you previously watched items with similar plot:
 1)  "scary movie" (ID: 3785) and gave rating: 3.0
 2)  "citizen kane" (ID: 923) and gave rating: 5.0
 3)  "shallow grave" (ID: 319) and gave rating: 4.0
```

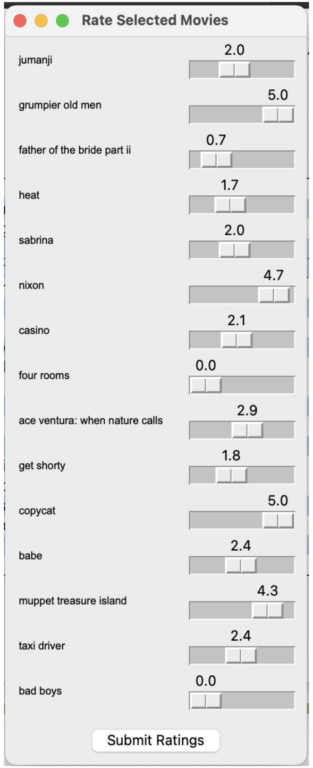
Evaluation of explanations - Individual Recommender

- **Innovation:** Subjective evaluation by using our user interface.

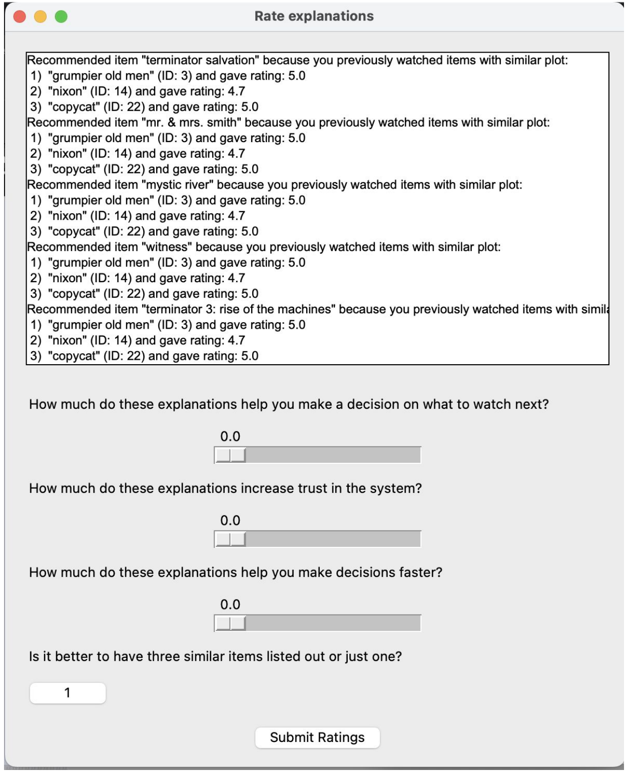


Pick at least 15 movies which you watched:

toy story
jumanji
grumpier old men
father of the bride part ii
heat
sabrina
goldeneye
nixon
casino
sense and sensibility
four rooms
ace ventura: when nature calls
get shorty
copycat
powder
leaving las vegas
dangerous minds
babe
dead man walking
clueless



jumanji	2.0
grumpier old men	5.0
father of the bride part ii	0.7
heat	1.7
sabrina	2.0
nixon	4.7
casino	2.1
four rooms	0.0
ace ventura: when nature calls	2.9
get shorty	1.8
copycat	5.0
babe	2.4
muppet treasure island	4.3
taxi driver	2.4
bad boys	0.0



Recommended item "terminator salvation" because you previously watched items with similar plot:
1) "grumpier old men" (ID: 3) and gave rating: 5.0
2) "nixon" (ID: 14) and gave rating: 4.7
3) "copycat" (ID: 22) and gave rating: 5.0
Recommended item "mr. & mrs. smith" because you previously watched items with similar plot:
1) "grumpier old men" (ID: 3) and gave rating: 5.0
2) "nixon" (ID: 14) and gave rating: 4.7
3) "copycat" (ID: 22) and gave rating: 5.0
Recommended item "mystic river" because you previously watched items with similar plot:
1) "grumpier old men" (ID: 3) and gave rating: 5.0
2) "nixon" (ID: 14) and gave rating: 4.7
3) "copycat" (ID: 22) and gave rating: 5.0
Recommended item "witness" because you previously watched items with similar plot:
1) "grumpier old men" (ID: 3) and gave rating: 5.0
2) "nixon" (ID: 14) and gave rating: 4.7
3) "copycat" (ID: 22) and gave rating: 5.0
Recommended item "terminator 3: rise of the machines" because you previously watched items with simila
1) "grumpier old men" (ID: 3) and gave rating: 5.0
2) "nixon" (ID: 14) and gave rating: 4.7
3) "copycat" (ID: 22) and gave rating: 5.0

How much do these explanations help you make a decision on what to watch next?

0.0

How much do these explanations increase trust in the system?

0.0

How much do these explanations help you make decisions faster?

0.0

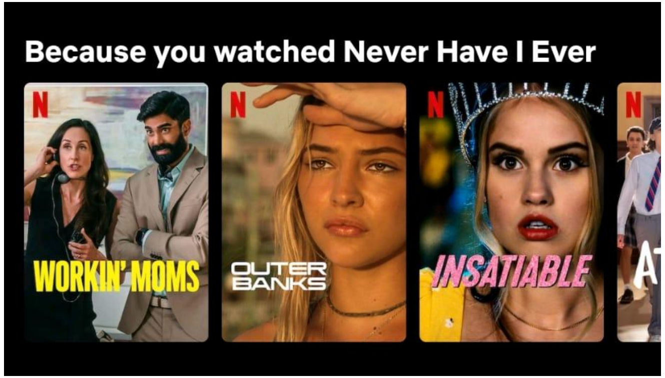
Is it better to have three similar items listed out or just one?

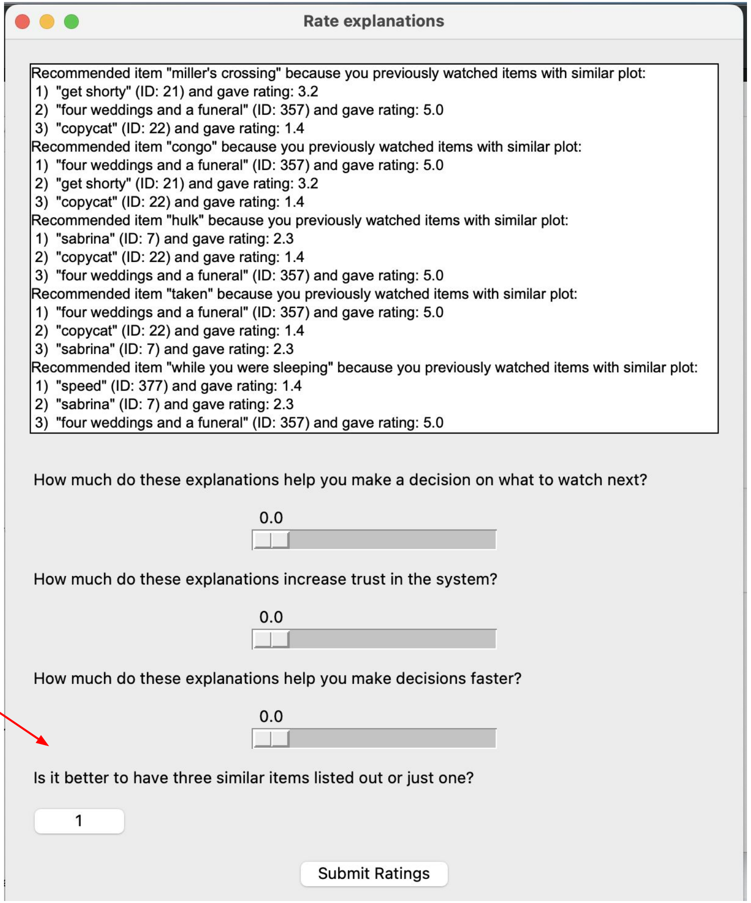
1

Evaluation of explanations - IR

Research questions:

1. How does presenting users with the first three most similar items, even if some have lower ratings, compared to the conventional approach of presenting only the highest-rated item (as seen in platforms like Netflix), impact users' satisfaction?





Recommended item "miller's crossing" because you previously watched items with similar plot:
1) "get shorty" (ID: 21) and gave rating: 3.2
2) "four weddings and a funeral" (ID: 357) and gave rating: 5.0
3) "copycat" (ID: 22) and gave rating: 1.4
Recommended item "congo" because you previously watched items with similar plot:
1) "four weddings and a funeral" (ID: 357) and gave rating: 5.0
2) "get shorty" (ID: 21) and gave rating: 3.2
3) "copycat" (ID: 22) and gave rating: 1.4
Recommended item "hulk" because you previously watched items with similar plot:
1) "sabrina" (ID: 7) and gave rating: 2.3
2) "copycat" (ID: 22) and gave rating: 1.4
3) "four weddings and a funeral" (ID: 357) and gave rating: 5.0
Recommended item "taken" because you previously watched items with similar plot:
1) "four weddings and a funeral" (ID: 357) and gave rating: 5.0
2) "copycat" (ID: 22) and gave rating: 1.4
3) "sabrina" (ID: 7) and gave rating: 2.3
Recommended item "while you were sleeping" because you previously watched items with similar plot:
1) "speed" (ID: 377) and gave rating: 1.4
2) "sabrina" (ID: 7) and gave rating: 2.3
3) "four weddings and a funeral" (ID: 357) and gave rating: 5.0

How much do these explanations help you make a decision on what to watch next?

0.0

How much do these explanations increase trust in the system?

0.0

How much do these explanations help you make decisions faster?

0.0

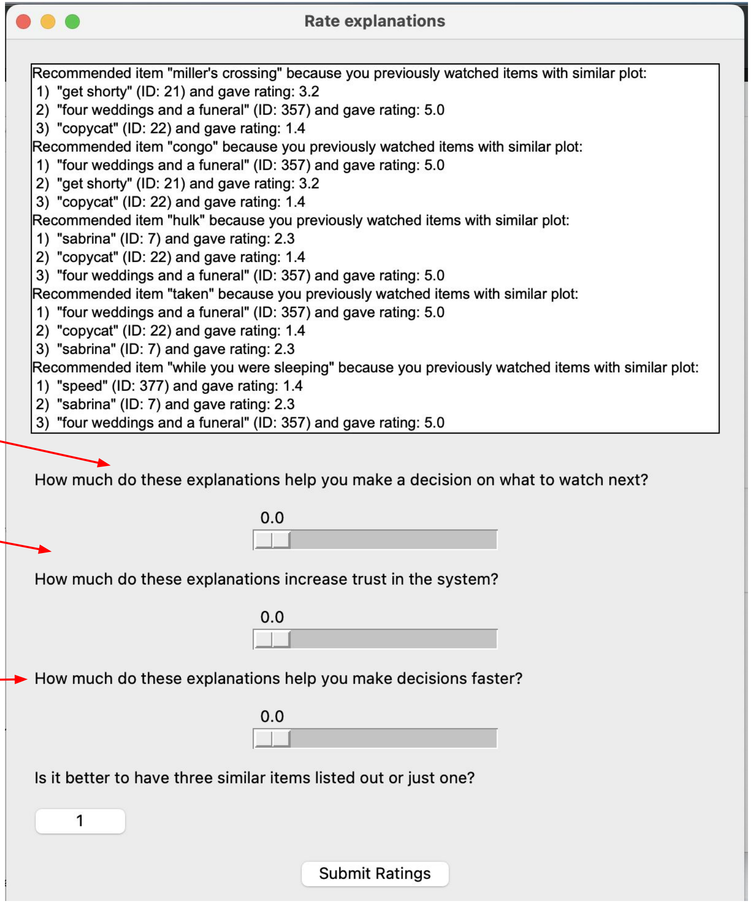
Is it better to have three similar items listed out or just one?

1

Evaluation of explanations - IR

Research questions:

2. How do users perceive the effectiveness of personalized explanations in enhancing their understanding of movie recommendations?
3. Do users trust the system more when they receive detailed explanations about how movie recommendations are generated?
4. How much do explanations contribute to the efficiency of users' decision-making processes while selecting movies?



Recommended item "miller's crossing" because you previously watched items with similar plot:
1) "get shorty" (ID: 21) and gave rating: 3.2
2) "four weddings and a funeral" (ID: 357) and gave rating: 5.0
3) "copycat" (ID: 22) and gave rating: 1.4
Recommended item "congo" because you previously watched items with similar plot:
1) "four weddings and a funeral" (ID: 357) and gave rating: 5.0
2) "get shorty" (ID: 21) and gave rating: 3.2
3) "copycat" (ID: 22) and gave rating: 1.4
Recommended item "hulk" because you previously watched items with similar plot:
1) "sabrina" (ID: 7) and gave rating: 2.3
2) "copycat" (ID: 22) and gave rating: 1.4
3) "four weddings and a funeral" (ID: 357) and gave rating: 5.0
Recommended item "taken" because you previously watched items with similar plot:
1) "four weddings and a funeral" (ID: 357) and gave rating: 5.0
2) "copycat" (ID: 22) and gave rating: 1.4
3) "sabrina" (ID: 7) and gave rating: 2.3
Recommended item "while you were sleeping" because you previously watched items with similar plot:
1) "speed" (ID: 377) and gave rating: 1.4
2) "sabrina" (ID: 7) and gave rating: 2.3
3) "four weddings and a funeral" (ID: 357) and gave rating: 5.0

How much do these explanations help you make a decision on what to watch next?

0.0

How much do these explanations increase trust in the system?

0.0

How much do these explanations help you make decisions faster?

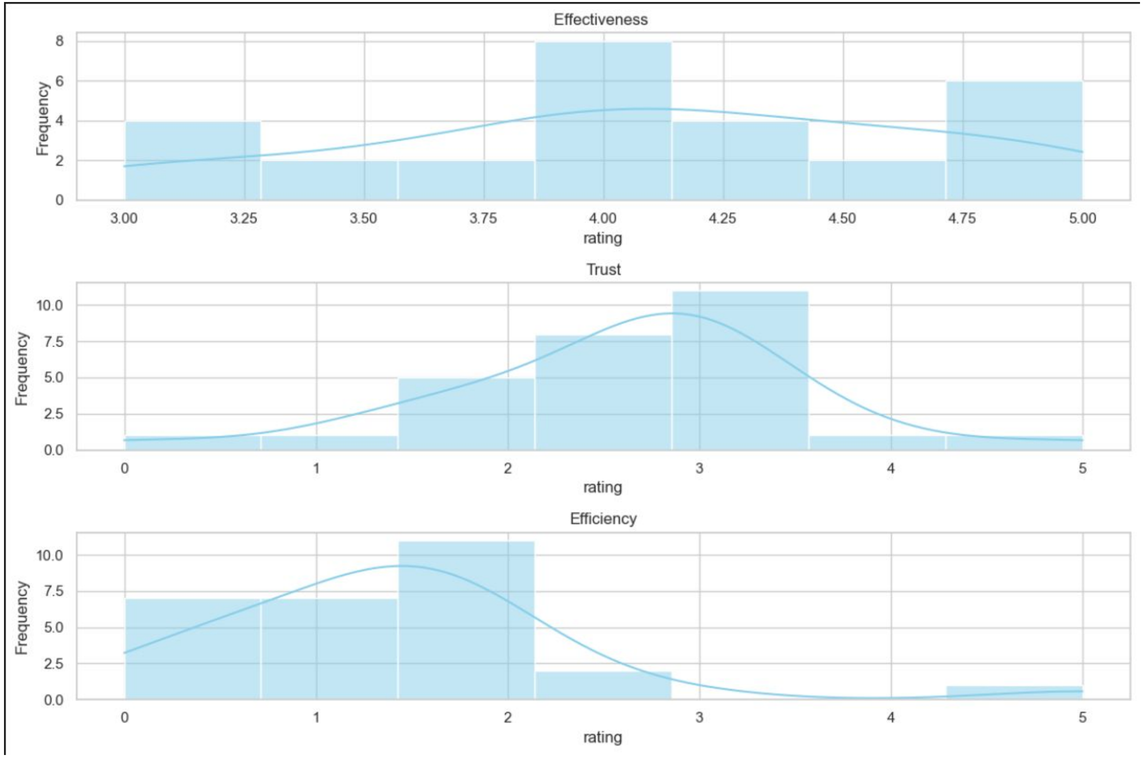
0.0

Is it better to have three similar items listed out or just one?

1

Evaluation of explanations (IR)- Results

- 28 participants (22 male, 6 female)
- In total around 82 % of users prefered having listed 3 similar items to each recommended item.
- Average rating for effectiveness: 4.08
- Average rating for trust: 2.61
- Average rating for efficiency: 1.41



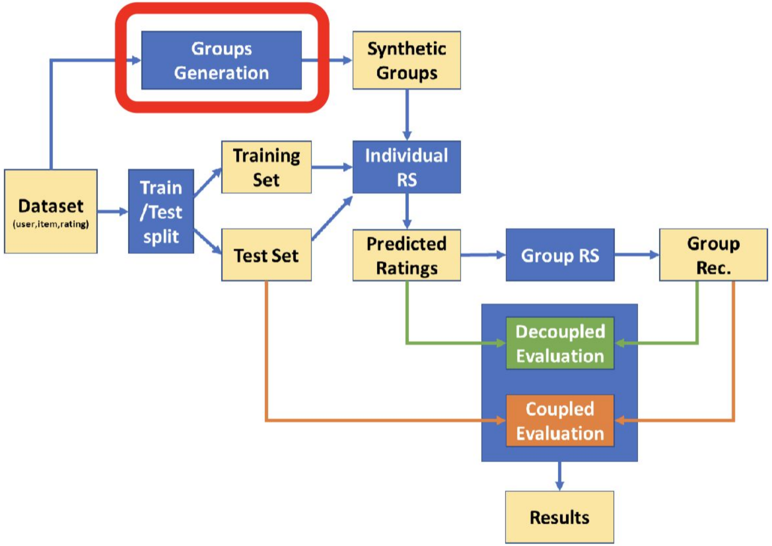
Group Recommender

- We fill in the user_matrix with predicted ratings in case of a non-rated movie.
- We do this to avoid a huge amount of sparsity when performing segmentation. We use content-based from IR.
- We use K-means (euclidean distance) to segment the data into 4 distinct groups based on their ratings.
    - In order to achieve a proper split, we run K-means 100 times, and select the result with the lowest inertia (sum of squared distances from each point to its assigned center).
    - We also explored k-medoids.
- We measure the average similarity between every pair of clusters. Used to test if k-means segmentation was working properly for different cluster sizes. 4 clusters was the most appropriate find.

Innovation: Using k-means to perform segmentation and then create synthetic groups.

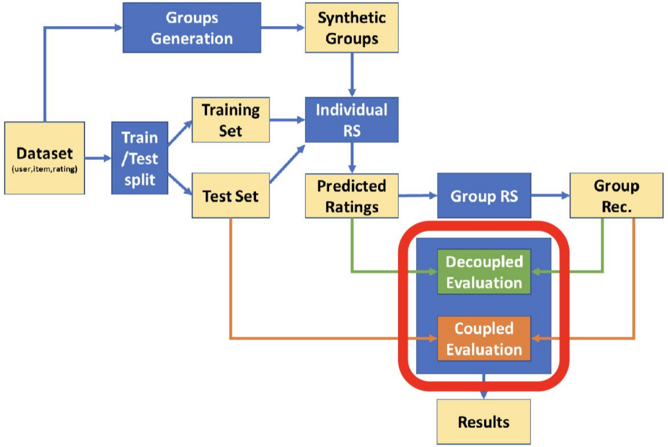
Group Recommender - Aggregation Strategies + GC

- Group composition strategies: Divergent, Uniform, Coalitional, and Minority.
- Additive Utilitarian, Approval Voting, Least Misery, Majority, Most Pleasure.



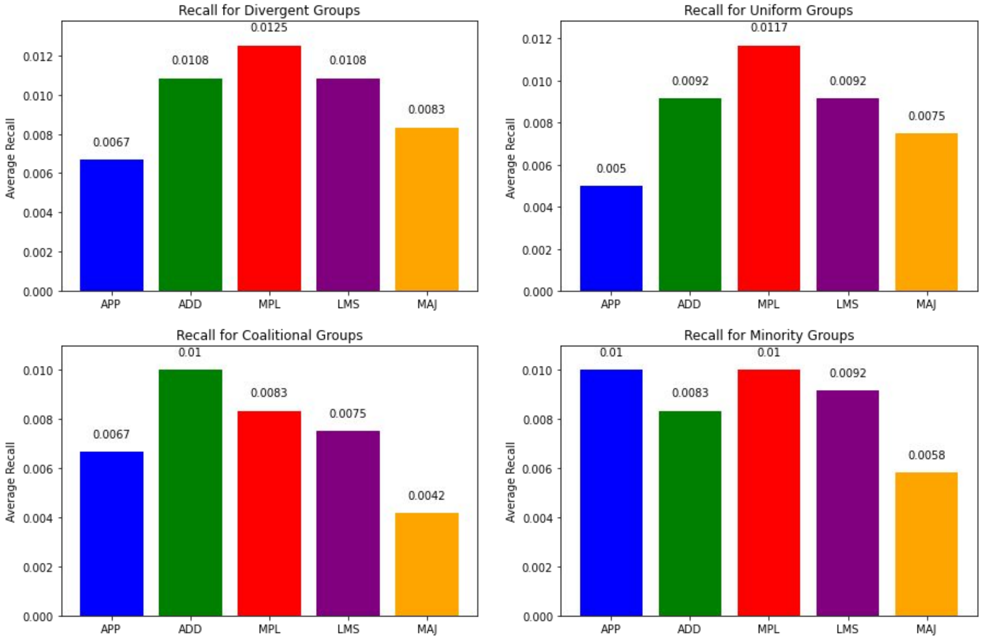
Evaluation - Group Recommender

- Decoupled evaluation, non-binarized feedback.
- Grountruth: User satisfaction - We will rank user's top 10 recommended movies and compare them with group's top 5 recommendations.
- All metrics are calculated with 1000 groups for each group composition strategy.
- We aim to evaluate the overall satisfaction of the group:
    - nDCG, precision, recall, F1-score.
- Satisfaction distribution among users:
    - Group fairness with discounted first hit (DFH).
- Implementing and evaluating several metrics for all group aggregations and compositions was the main action to mitigate DEB.



Evaluation Results- Group Recommender

- MPL seems to perform well overall, especially in divergent and uniform groups.
- MAJ is the overall worst performer.
- APP seems to be the best in minority groups.
- ADD is the best for coalitional.
- A high recall means that the system is good at suggesting most of the items that the user would find relevant.


Recall for Divergent Groups

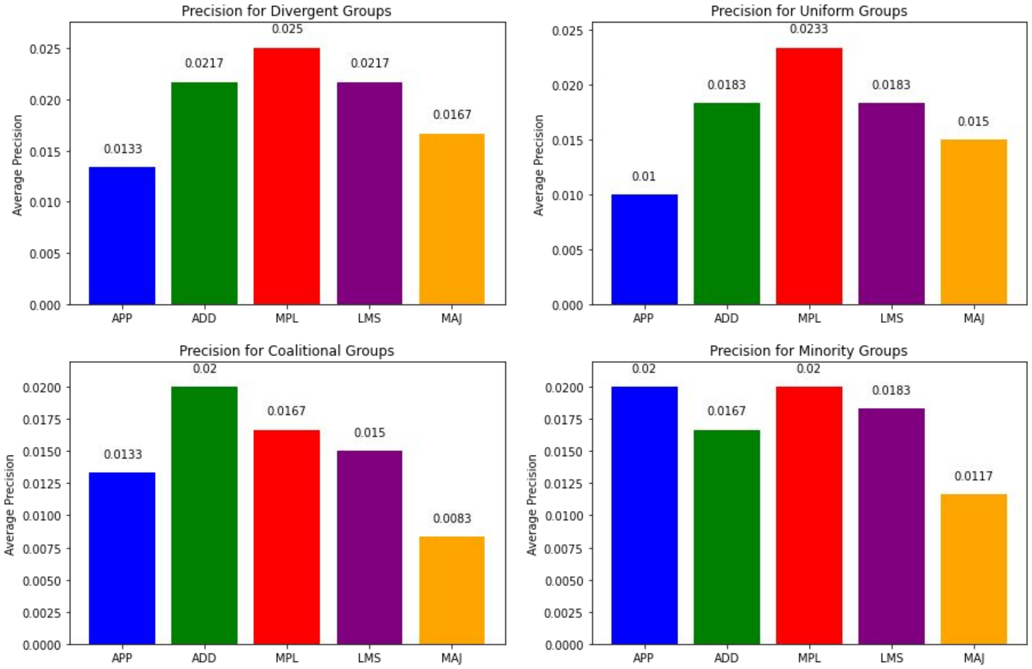
Recall for Uniform Groups

Recall for Coalitional Groups

Recall for Minority Groups

Evaluation Results- Group Recommender

- Similar distribution to recall.
- Balanced recommendations.



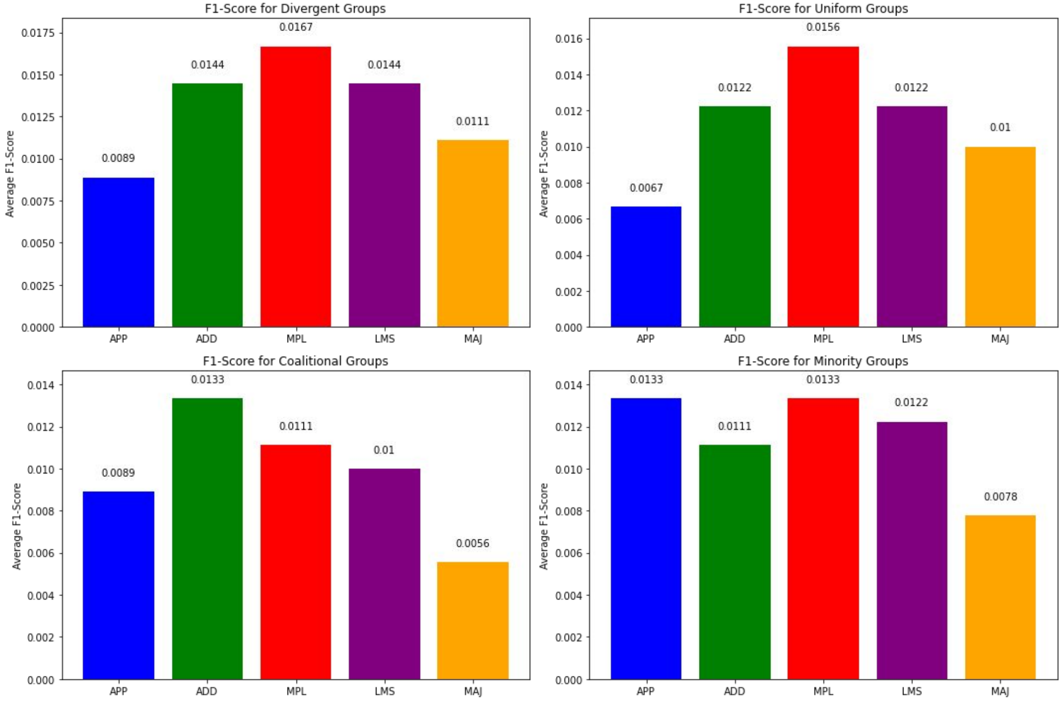
Precision for Divergent Groups

Precision for Uniform Groups

Precision for Coalitional Groups

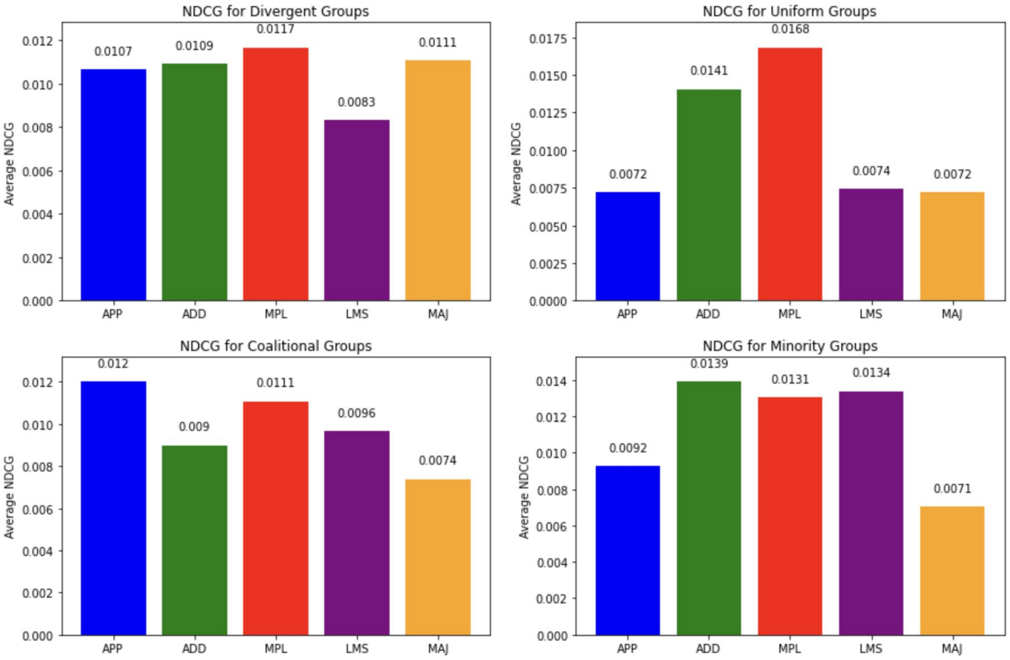
Precision for Minority Groups

Evaluation Results- Group Recommender



Evaluation Results- Group Recommender

- Best nDCG overall goes to MPL aggregation.
- Significantly higher than others for divergent and uniform groups.



NDCG for Divergent Groups

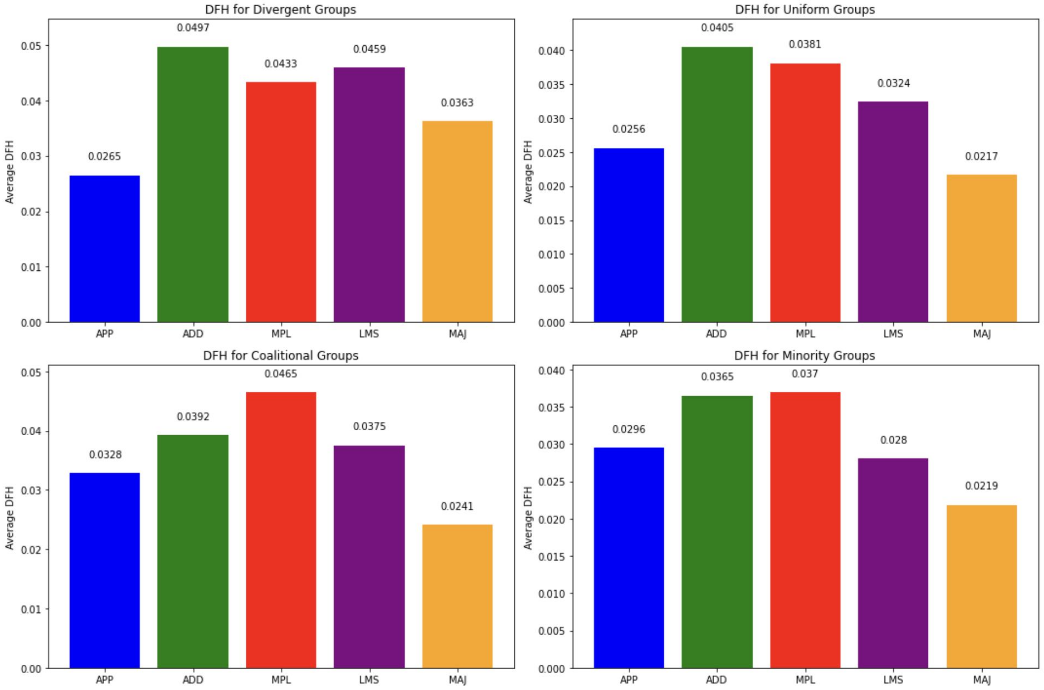
NDCG for Uniform Groups

NDCG for Coalitional Groups

NDCG for Minority Groups

Evaluation Results- Group Recommender

- ADD seemed to be the most fair aggregation strategy. Followed by MPL.
- Perhaps due to high diversity in the dataset.



Explanations - Group Recommender

- Goal of the explanations is to provide additional information that is associated with the recommendations. This is done to achieve goals such as promoting consensus, increasing transparency, effectiveness, usability and user satisfaction among users.
- The explanations provided are done based on social choice aggregation. These explanations are focused on reassuring users about the quality and relevance of the recommended movies, and potentially help repair any mistrust or confusion about the system's functionality or rationale.
- Format based on Barile et al, 2023 paper, but we aim to innovate evaluation of the explanations by performing subjective evaluation:

No explanation - "We advise the group to consider movie x."

Basic explanation - "Movie x has been recommended to the group because it holds the maximum cumulative rating."

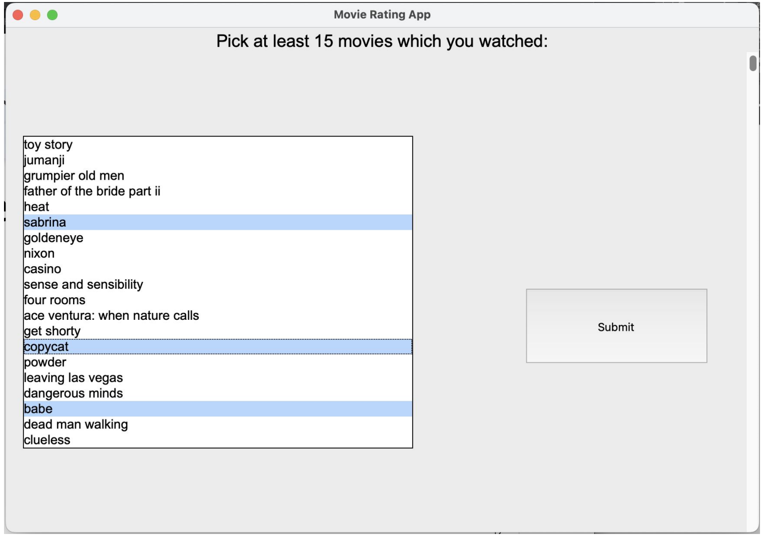
Detailed explanation -"Movie x has been recommended to the group because it holds the maximum cumulative rating. This means after adding all group members ratings' for all movies, movie x had the highest sum."

Evaluation of explanations - Group Recommender

- How do the three types of explanations (No explanation, Basic, Detailed) influence users’ fairness perception, consensus perception, or satisfaction?

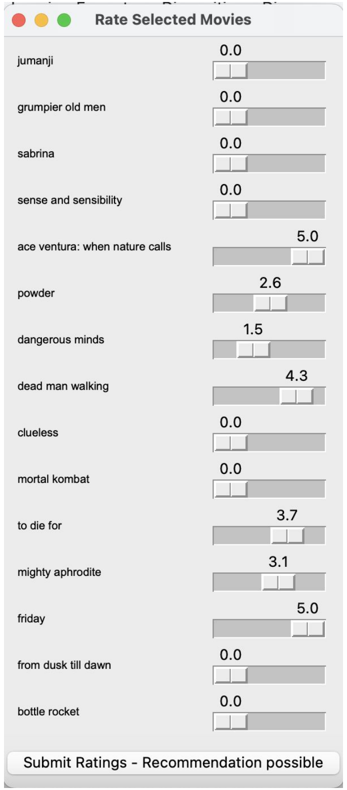
- User fills in the data (at least 15 movies and rates them)
    - Add the user to the database of new users.
- We make groups of 4.
- Random assign the user to an explanation type out of 3.
- Ask questions:
    - **Fairness Perception**: "Did you feel the recommendations were fair to all group members?"
    - **Consensus Perception**: "Did the explanation help your group reach a consensus?"
    - **Satisfaction**: "Were you satisfied with the recommendations provided?

Evaluation of explanations - Group Recommender

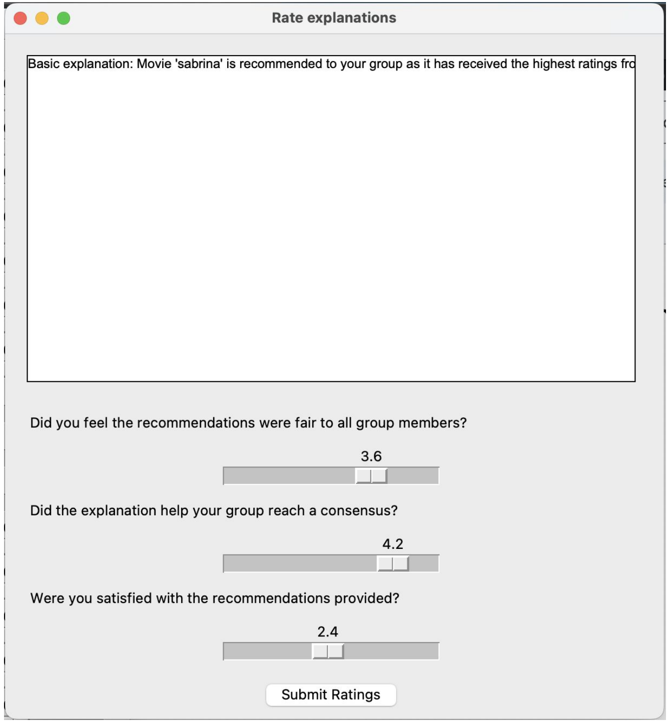

Movie Rating App

Pick at least 15 movies which you watched:

toy story
jumanji
grumpier old men
father of the bride part ii
heat
sabrina
goldeneye
nixon
casino
sense and sensibility
four rooms
ace ventura: when nature calls
get shorty
copycat
powder
leaving las vegas
dangerous minds
babe
dead man walking
clueless


Rate Selected Movies

jumanji 0.0
grumpier old men 0.0
sabrina 0.0
sense and sensibility 0.0
ace ventura: when nature calls 5.0
powder 2.6
dangerous minds 1.5
dead man walking 4.3
clueless 0.0
mortal kombat 0.0
to die for 3.7
mighty aphrodite 3.1
friday 5.0
from dusk till dawn 0.0
bottle rocket 0.0


Rate explanations

Basic explanation: Movie 'sabrina' is recommended to your group as it has received the highest ratings fro

Did you feel the recommendations were fair to all group members? 3.6

Did the explanation help your group reach a consensus? 4.2

Were you satisfied with the recommendations provided? 2.4

Evaluation of explanations - Group Recommender

- **28 participants (22 male, 6 female)**
- **7 groups of 4 people each**

	None	**Basic**	**Detailed**
Average rating for fairness:	3.13	3.94	3.76
Average rating for consensus:	0.71	2.92	3.51
Average rating for user satisfaction:	1.78	1.94	1.92

Conclusion

- Best individual recommender is the Content-based based on RMSE and MAE results.
- Explanations show that users prefer 3 similar items displayed instead of 1.
- Evaluation of the explanations demonstrate that the explanations are significantly effective when it comes to making a decision. Trust seems to slightly increase according to the results for some of the users, whereas the results for efficiency show that explanations mostly don’t affect on the time it takes to make the decision.
- MPL strategy in divergent and uniform groups performs best.
- More detailed explanations can help a group reach a higher consensus perception.

Appendix - basic.popular

- The mentioned algorithm will recommend the items, movie in this case, ordered by popularity.
- What are the pros and cons of the first baseline, on our MovieLens dataset?
    - It is a basic algorithm, easy to scale for a big dataset like ours, even if we use small samples.
    - It will be our first comparison point to more advanced solutions.
    - It is a non personalized algorithm so unfortunately will not take into consideration users preferences, but will be interesting to compare with personalized baselines.

Appendix - bias.Bias()

- We implemented another baseline in order to compare with the popularity one, which did not perform that well. The new baseline we will implement is BiasOnly model.
- The pros and cons for this algorithm and why we chose it are:
  - As being simple algorithm, is again suitable for large and sparse datasets as our
  - In comparison with the popularity one this is personalized taking into account item and user biases.

Appendix - Fallback Algorithm

- Now let's combine the two created baselines and use the fallback algorithm which will take as parameters both the Popularity and Bias algorithms.
- The idea here is that we will use our Bias algorithm first and if it fails to generate recommendations, the fallback method will then use Popularity algorithm.
- In such case recommendations might be combined so we will expect better diversity and at the end, basically better results. We will see if this hypothesis holds.
- The fallback method will have increased complexity but this is fine as we only work with a small sample od the data.
- One advantage of adding popularity as a fallback to bias might be "the cold start problem". On the other hand, we might be more prone to overfitting so it can in some situations obtain a larger error.