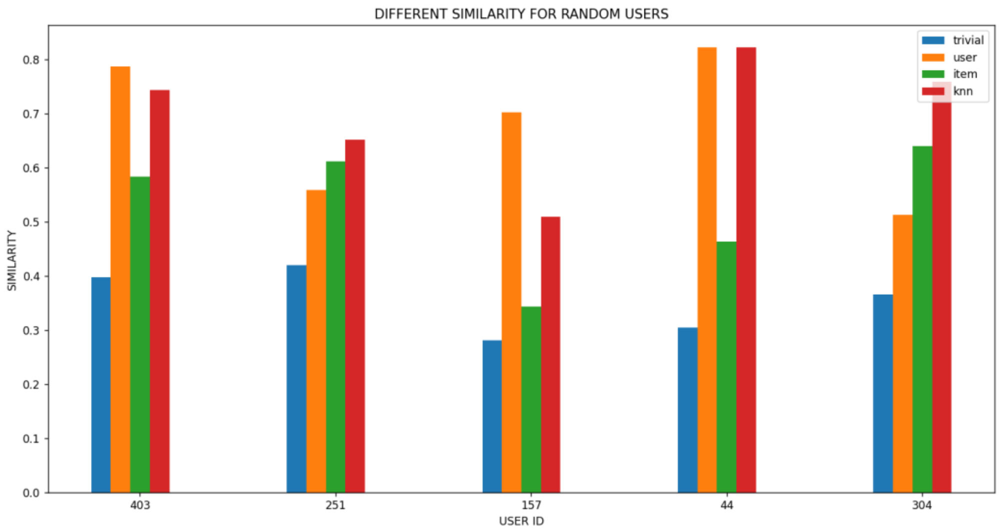
1 Evaluation criteria

The following criteria should be used to evaluate this practice:

1. Complete the implementation of trivial recommender. (0.5 pts)
2. Complete the implementation of user-based and item-based recommender. Previously, should have implemented the similarity code, and ensured to work correctly. (3 pts)
3. Complete the implementation of K-NN hybrid recommender. It's possible to use others combination metrics, but it's necessary to ensure that the results of user-based and item-based are normalized. (1 pts)
4. Complete the validation code. Verify that the TF-IDF and similarity codes are correct (a simple test is to compute the similarity of a list of recommendations with itself, the similarity must be 1). (2 pts)
5. For each recommender system and validation, explain how they have been implemented, including relevant details about the algorithms, data structure, metrics used, etc. (2 pts)
6. Create some graphs to visualize the different similarities obtained by the models and compare them appropriately explaining which model works better, which worse, and why. (1.5 pts)

2 Possible experimental result



In the graph above, the similarity calculations between genres obtained by the different models and those of the films rated by the user within the validation set are reflected for each user.

We can observe that the knn-based recommender system is the best one because it combines the information of users and items to compute the predictions. Moreover, the user-based model computes recommendations with high precision, the similarity between users is sophisticated and robust because in the dataset there are some users who have rated many movies, and these users have a significant weight when computing the similarity.

The item-based recommender has a regular precision, is probably that many movies have been rated by few users and this would affect the computation of similarity between items.

We can observe that the trivial recommender is the worst because it only provides recommendations based on the highest average ratings to users, without offering personalized recommendations for each user.

Finally, we have analyzed that both knn-based and user-based models generate high precision for these random users, but to check which one is the best, it's necessary to compare the different similarities obtained by models for all users in dataset.