**AIE425 Intelligent Recommender Systems, Fall Semester 24/25**  
**Assignment #2: Significance Weighting-based Neighborhood CF Filters**

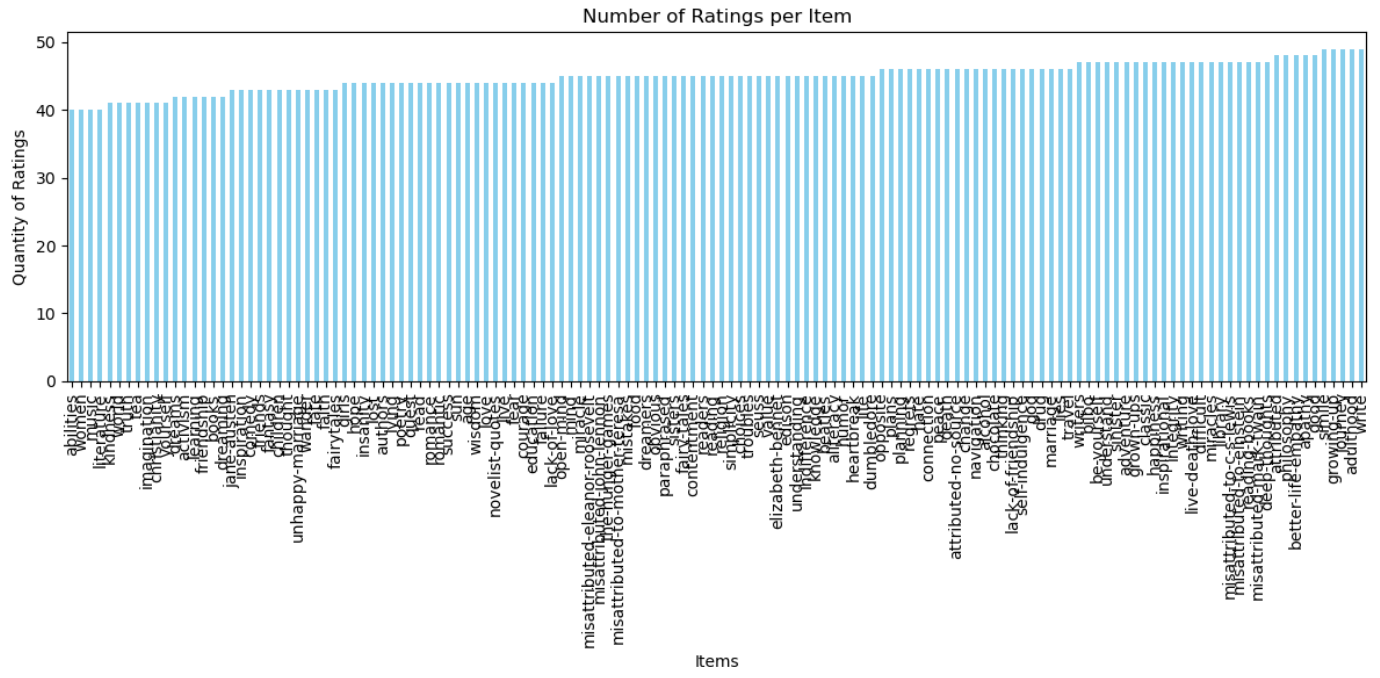
**Student ID:** A20001136  
**Full Name:** Abdelrahman El-Sayyad

**Table of Contents**

1. Outcomes of Section 3.1
2. Summary of the Comparison of Part 1 and Part 2
3. Conclusion

**Outcomes of Section 3.1**

In this section, we prepared and analyzed our dataset according to the requirements listed in Section 3.1.

1. **Use of Dataset and Rating Scale (Requirements 3.1.1 & 3.1.2):**  
   I used the dataset created in Assignment 1. The ratings in the dataset were scaled to a 1-to-5 scale as needed.
2. **Counting Total Number of Users and Items (3.1.3 & 3.1.4):**  
   After loading the dataset, we counted the total number of users (tnu) and the total number of items (tni). We stored the values in the variables tnu and tni. For this assignment, we found tnu = 50 users and tni = 138 items.
3. **Counting the Number of Ratings per Product (3.1.5):**  
   I calculated the number of ratings for each product in the dataset. The rating count for each item was shown, which confirmed that every item in our pre-processed dataset had a known number of ratings before missing values were introduced.
4. **Selecting Active Users and Introducing Missing Ratings (3.1.6):**  
   For the choice of active users, we picked three: U1, U2, and U3. We introduced 2 missing ratings in one, 3 missing ratings in the other, and 5 missing ratings in the last in carefully chosen items so we had active users at various missing data levels.
5. **Selecting Target Items with Missing Percentages (3.1.7):**  
   I identified two items as target items, namely I1 and I2. In I1, we introduced 4% missing ratings, while in I2, we introduced 10% missing ratings. These target items would serve as test cases for item-based predictions.
6. **Co-Rating Analysis and 2-D Array (3.1.8 & 3.1.9):**  
   I have done a co-rating analysis to count how many users co-rated items with each active user (No\_common\_users) and how many co-rated items there were (No\_coRated\_items). Then we created a 2-D array where the first column listed No\_common\_users in descending order and the second column listed the corresponding No\_coRated\_items.
7. **Plotting the Ratings Count per Item Curve (3.1.10):**  
   I created a bar chart—a sort of curve—representing how many ratings per item were available within our dataset. This, in turn, helped us capture the variability of rating presence across items.
8. **Determining the Threshold β (3.1.11):**  
   I calculated threshold β for each active user by identifying the maximum number of users who co-rated at least 30% of the items with each active user. This value of β would later be used in applying the discount factor (DF).
9. **Saving and Displaying Values (3.1.12):**  
   All intermediate values, including tnu, tni, co-rating arrays, β thresholds, and predictions, were either displayed or saved as part of the final results. We also saved the modified dataset.

**Summary of the Comparison of Part 1 and Part 2**

**Part 1 – (User-Based CF):**  
In Part 1, I discussed the use of different similarity measures and weighting schemes to build a user-based Collaborative Filtering:

* **Case Study 1.1 (Cosine without Mean-Centering):**  
  I computed user-user similarities by using cosine similarity directly on the ratings without mean centering. Then, predict missing ratings for active users after selecting the top 20% closest users. Introduce a discount factor based on threshold β and calculate discounted similarities. Observe the change in top neighbors and predictions.
* **Case Study 1.2 (Cosine with Mean-Centering):**  
  We accounted for user bias by mean-centering the user ratings. The cosine similarity computed on mean-centered data gave a different similarity landscape. Applying DF again refined the top neighbors and predictions. Compared to the non-mean-centered approach, mean-centering usually resulted in more balanced and potentially more accurate predictions.
* **Case Study 1.3 (PCC with Mean-Centering):**  
  The PCC inherently provided a similarity measure based on mean ratings. The introduction of DF here influenced which neighbors were considered "significant" and improved the reliability of predictions for the active users. The scores of similarities became subtler since the PCC measures the linear correlation instead of just the angular similarity.

Comparing Case Studies 1.1, 1.2, and 1.3, we found that mean-centering and the use of PCC generally provided enhancements in the identification of truly similar users.  
The application of DF further enhanced these similarities by highlighting users who had co-rated a significant number of items. Therefore, in Part 1, both mean-centering and DF tended to provide more reliable user-based recommendations.

**Part 2 (Item-Based CF):**  
Part 2 repeated the steps of Part 1, but for item-based CF:

* **Case Study 2.1 (Cosine without Mean-Centering):**  
  I calculated item-item similarities and estimated the top 25% closest items. After that, we predicted missing ratings for target items. The introduction of DF refined the set of similar items and enhanced the quality of predictions.
* **Case Study 2.2 (Cosine with Mean-Centering):**  
  Mean-centering in item-based CF helped account for item-specific rating biases. With the application of DF, again, we observed improvements in highlighting truly significant item-item relationships leading to better predictions.
* **Case Study 2.3 (PCC with Mean-Centering):**  
  Using PCC for item-item similarity captured more subtle correlations between items’ rating patterns. Applying DF further enhanced the results. Comparisons between item-based approaches showed that mean-centering and PCC, along with DF, improved the precision of item-based recommendations.

**Comparing Part 1 (user-based) and Part 2 (item-based), we found:**

* Both user-based and item-based CF benefited from mean-centering and from applying the discount factor to refine similarity measures.
* Cosine similarity was simpler but less sensitive to shifts in rating scale than PCC.
* PCC, when combined with mean-centering, often yielded more intuitively meaningful similarities, while DF ensured that neighbors/items with insufficient co-ratings were de-emphasized.

In other words, DF and centering enhanced the top-N user/item choice and ensuing rating prediction quality for both user- and item-based CF.

**Conclusion**

Through this assignment, we explored various aspects of user-based and item-based collaborative filtering, including:

* Adjusting ratings to a 1-to-5 scale and understanding the distribution of ratings.
* Introducing missing values to mimic real-world data sparsity.
* Identifying active users and target items with controlled patterns of missingness.
* Computing similarities using Cosine and PCC measures, with and without mean-centering.
* Applying a discount factor (DF) based on a threshold β to emphasize neighbors or items that co-rated a significant portion of items.
* The experiments showed that mean-centering ratings often led to improved similarity computations, making them more robust to individual biases. Using PCC generally provided a more correlation-based perspective, often yielding improved neighbor quality. Applying the discount factor further refined the sets of neighbors/items, ensuring that only those with a substantial amount of shared rating information influenced the predictions.
* Overall, these adjustments (mean-centering, PCC, and DF) had a noticeable positive impact on recommendation performance. They improved the reliability of similarity measures and enhanced the accuracy of predictions. From a practical standpoint, incorporating significance weighting (DF) and bias adjustments (mean-centering) would likely lead to more trustworthy and stable recommender systems.
* In conclusion, the significance weighting-based neighborhood CF filters enhanced both user-based and item-based recommendations, demonstrating that careful consideration of similarity measures, bias adjustments, and weighting schemes is crucial for building high-quality recommendation systems.

**End of Report**