AIE425 Intelligent Recommender Systems, Fall Semester 24/25

Assignment #1: Neighborhood CF models (user, item-based CF)

221101235, Marina Reda Abdullah Mekhael

Table of Contents (TOC):

Intro / Core idea :

I will explore and apply on two collaborative filtering approaches for making recommendations: user-based and item-based. User-based filtering focuses on finding similar users to predict ratings, while item-based filtering identifies similarities between items. Both methods have their strengths where user-based filtering excels at leveraging user preferences, while item-based filtering effectively highlights item relationships. The goal of assignment is to understand how these strategies can be utilized to improve user satisfaction through better-tailored suggestions.

Assignment Requirements:

1. Search for suitable companies in various domains that use recommender systems (see lecture 1 & 2).

Answer:

E-commerce: Amazon, Shopee, Lazada, Taobao

Travel: Booking.com, Expedia, Airbnb

Point-of-Interest: Yelp, Foursquare, Groupon

Multi-Media:

- Video: YouTube, Netflix, IMDb
- Social Network: Facebook, Twitter, LinkedIn
- Music: Spotify, Pandora, QQ Music
- News: CNN, BBC News, New York Times
- Photo: Pinterest, Flickr, Instagram

2. List them in your report and choose one or more as the data source for the assignment.

Answer : I’ve chosen Amazon as the data source.

3. Describe how the chosen company collects customer feedback and what rating type is used.

Answer: Amazon collects customer feedback through a 1 to 5 star rating system and it is interval and explicit ratings where each number reflects an increasing level of satisfaction.

4. Prepare the collected data and take the necessary preprocessing procedures to clean it and express the feedback in the form of integer values.

Answer: Done , look at next answer

5. Explain clearly the process you used to obtain and preprocess data, as well as the rating type.

Answer: I used Web Scraping and I collected data on: User ID, Product ID and Rating (1 to 5 scale- interval ratings), I cleaned it by removing duplicate entries and convert id to users and items to simple names and I have chosen the five best users(common set) who jointly voted on the same products to work on it Because there are many empty values, which means that they do not share the same taste, then reshaped a user-item matrix, where each row represented a user, each column represented a product, and the cells contained the corresponding ratings.

6. Create your own user-item matrix and use it as the dataset for this assignment.

Answer:

product_id	treadmill	dumbbells	yoga_mat	elliptical	kettlebell
User1	5	3	0	4	4
User2	0	4	3	3	2
User3	4	0	5	4	3
User4	0	2	1	3	4
User5	3	0	4	0	5

7. Give a complete description of the created dataset.

Answer: The dataset includes five users and five sports equipment items (treadmill, dumbbells, yoga mat, elliptical, kettlebell). Each user has rated some of the items on a scale from 1 to 5, where higher ratings mean stronger preference, while a "0" means the user hasn’t rated that item.

8. Compute the average rating and copy the results into your report under the "Assignment Results" section.

Answer: Done

9. Give a complete background/overview about user-based and item-based CF algorithms, and their detailed analytical solutions.

Answer: Collaborative Filtering (CF) is widely used in recommendation systems, relying on past user interactions, such as ratings. CF mainly uses two approaches: User-Based CF and Item-Based CF.

1. User-Based Collaborative Filtering (User-CF): User-based CF recommends items to a user based on the preferences of similar users. Here’s how it works: First, create a user-item matrix where each row represents a user, each column an item, and the cells contain ratings. Then, calculate similarity between users using methods like: **Cosine Similarity:** Measures the cosine angle between two users' rating vectors. It indicates how closely their tastes align and **Pearson Correlation Coefficient:** Measures the linear relationship between users’ ratings, considering both similarity in direction and magnitude.

After identifying "similar users" (neighbors), predict ratings for items the target user.

2. Item-Based Collaborative Filtering (Item-CF): Item-based CF finds relationships between items rather than users. If two items are often rated similarly by many users, they're likely related. This method recommends items similar to what the user has liked before. the process: Use a user-item matrix as before. Then, Calculate item similarity, also using Cosine Similarity or Pearson Correlation, by comparing rating patterns across users. After that recommend items similar to those they liked by averaging ratings from similar items.

   Detailed Analytical Solutions:

$$cosine(\overrightarrow{r_u}, \overrightarrow{r_v}) = \frac{\sum_{u \in U}(r_{u,a})(r_{u,b})}{\sqrt{\sum_{u \in U}(r_{u,a})^2}\sqrt{\sum_{u \in U}(r_{u,b})^2}}$$

$$Pearson(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

$$AdjustedCosine(i,j) = \frac{\sum_{u \in U_i \cap U_j}(r_{u,i} - \overline{r_u})(r_{u,j} - \overline{r_u})}{\sqrt{\sum_{u \in U_i \cap U_j}(r_{u,i} - \overline{r_u})^2}\sqrt{\sum_{u \in U_i \cap U_j}(r_{u,j} - \overline{r_u})^2}} = \frac{\sum_{u \in U_i \cap U_j} S_{u,i} \cdot S_{u,j}}{\sqrt{\sum_{u \in U_i \cap U_j}(S_{u,i})^2}\sqrt{\sum_{u \in U_i \cap U_j}(S_{u,j})^2}}$$

10. Compute the similarity using both the cosine similarity measure and the Pearson correlation coefficient to identify a peer group of users in the case of user-based CFs and a peer group of items in the case of item-based CFs.

Answer: Cosine Similarity Between Users :

$$cosine(\overrightarrow{r_u}, \overrightarrow{r_v}) = \frac{\sum_{u \in U}(r_{u,a})(r_{u,b})}{\sqrt{\sum_{u \in U}(r_{u,a})^2}\sqrt{\sum_{u \in U}(r_{u,b})^2}}$$

Cosine(1,3)= [(4×5)+(4×4)+(3×4)]/[[(25+16+16)^0.5]*[(16+16+9)^0.5]]=[20+16+12]/[(57^0.5)*(41^0.5)]=0.993

Cosine(2,3)= [(3×5)+(3×4)+(2×3)]/[[(9+9+4)^0.5]*[(25+16+9)^0.5]]= 0.995

Cosine(4,3)= [(1×5)+(3×4)+(3×4)]/[[(25+16+9)^0.5]*[(16+1+9)^0.5]]= 0.804

Cosine(5,3)= [(4×3)+(5×4)+(3×5)]/[[(25+16+9)^0.5]*[(25+16+9)^0.5]]=0.94

Pearson correlation coefficient for user based:

$$Pearson(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

	treadmill	dumbbells	yoga_mat	elliptical	kettlebell	Mean rate
User1	1	-1	?	0	0	4
User2	?	1	0	0	-1	3
User3	0	?	1	0	-1	4
User4	?	-0.5	-1.5	0.5	1.5	2.5
User5	-1	?	0	?	1	4

Pearson(1,3)= (1*0)+(0*0)+(0*-1)/[[(1)^0.5]*[(1)^0.5]]= 0.00

Pearson(2,3)=(-1*-1)/[[(1)^0.5]*(2)^0.5]= 0.707

Pearson(4,3)=(1*-1.5)+(-1*1.5)/[[(2)^0.5]*[(19)^0.5/2]]= 0.729

Pearson(5,3)=(-1*1)/[[(2)^0.5]*[(2)^0.5]]= -0.5

Cosine Similarity for item based:

$$AdjustedCosine(i,j) = \frac{\sum_{u \in U_i \cap U_j}(r_{u,i} - \overline{r_u})(r_{u,j} - \overline{r_u})}{\sqrt{\sum_{u \in U_i \cap U_j}(r_{u,i} - \overline{r_u})^2}\sqrt{\sum_{u \in U_i \cap U_j}(r_{u,j} - \overline{r_u})^2}} = \frac{\sum_{u \in U_i \cap U_j} S_{u,i} \cdot S_{u,j}}{\sqrt{\sum_{u \in U_i \cap U_j}(S_{u,i})^2}\sqrt{\sum_{u \in U_i \cap U_j}(S_{u,j})^2}}$$

mean-centered ratings matrix:

	treadmill	dumbbells	yoga_mat	elliptical	kettlebell	Mean rate
User1	1	-1	?	0	0	4
User2	?	1	0	0	-1	3
User3	0	?	1	0	-1	4
User4	?	-0.5	-1.5	0.5	1.5	2.5
User5	-1	?	0	?	1	4

AdjustedCosine(i2,i1)=(1*-1)/[[(1)^0.5]*[(1)^0.5]]= -1

AdjustedCosine(i2,i3)=(-0.5*-1.5)/[[(1+(-0.5)^2)^0.5]*[(-1.5^2)^0.5]]= 0.447

AdjustedCosine(i2,i4)=(-0.5*0.5)/[[(1+1+0.25)^0.5]*[(0.25)^0.5]]= -0.333

AdjustedCosine(i2,i5)=(1*-1)+(-0.5*1.5)/[[(1+1+0.25)^0.5]*[(1+2.25)^0.5]]=-0.647

11. Compare the results of measuring similarity using similarity measure with the results using Pearson correlation coefficient and emphasizing the pros and cons of each technique.

Answer: The highest similarity scores were observed between Users 1 & 3 (0.993) and Users 2 & 3 (0.995), indicating that these users have similar rating patterns where Cosine Similarity: This measure focuses on the *angle* between rating vectors, not their magnitude. It's beneficial when the absolute rating values matter less than their relative pattern. Pros: It's simple to calculate and works well for comparing users/items with different rating scales. Cons: It ignores average user ratings, so it can misrepresent similarity if users rate consistently higher or lower than others.

This measure showed that Users 2 & 3 (0.707) and Users 4 & 3 (0.729) are somewhat similar but with weaker correlations than seen in cosine similarity where Pearson Correlation Coefficient: This measure considers how ratings *deviate from the mean*, making it effective for handling user biases. Pros: It accounts for user/item average ratings, so it provides a more balanced view of similarity in cases where users have distinct rating habits. Cons: It assumes a linear relationship and may be more complex to compute due to mean adjustments.

12. Copy the results into your report under the "Assignment Results" section.

Answer: Done.

13. Compute the rating prediction and the top-N list of recommended users/products in case of user-based and item-based CF, each case must be performed using the cosine similarity measure and the Pearson correlation coefficient.

Answer: Cosine Similarity Rating Prediction (User-Based CF):

$$predicted\ rating = \frac{\sum_{b \in N} sim(a,b) * (r_{b,p})}{\sum_{b \in N} sim(a,b)}$$

The highest similarity scores were observed between Users 1 & 3 (0.993) and Users 2 & 3 (0.995)

Pred(3,2)=(3* 0.993)+(4* 0.995)/0.993+ 0.995=3.5

Pearson Correlation Rating Prediction (User-Based CF):

$$pred(a,p) = \overline{r_a} + \frac{\sum_{b \in N} sim(a,b) * (r_{b,p} - \overline{r_b})}{\sum_{b \in N} sim(a,b)}$$

measure showed that Users 2 & 3 (0.707) and Users 4 & 3 (0.729) are similar

Pred(3,2)= 4+[[(1* 0.707)+(-0.5* 0.729)]/0.707+ 0.729]=4.238

Cosine Similarity Rating Prediction (Item-Based CF):

The highest similarity scores were observed between item 2 & 3 (0.447) dumbbells and yoga_mat

$$pred(u,t) = \frac{\sum_{j \in Q_t(u)} AdjustedCosine(j,t) \cdot r_{u,j}}{\sum_{j \in Q_t(u)} |AdjustedCosine(j,t)|}$$

Pred(3,2)=5* 0.447/0.447=5

Pred(5,2)=4* 0.447/0.447=4

14. Compare the results of the rating prediction and the top-N list of recommended users/products after performing step 14.

	Prediction for User 3 on Item 2	Top-N Recommendations
User-Based CF (Cosine)	3.5	recommend items based on highly similar users' preferences. v1,v2 to u3
User-Based CF (Pearson)	4.238	recommend items based on highly similar users' preferences.v2,v4 to u3
Item-Based CF (Cosine)	5 (User 3), 4 (User 5)	Recommend yoga mat i3 based on dumbbells i2

15. Copy the results into your report under the "Assignment Results" section.

Answer: Done

16. Present, describe, compare, and evaluate the results in all cases.

Answer: User-Based Collaborative Filtering: Cosine Similarity: Users 1 and 3 (0.993) and Users 2 and 3 (0.995) had the highest similarity scores, which made their ratings important in predicting ratings for User 3. This method gave predictions close to actual ratings, as it emphasized similar user preferences. And Pearson Correlation: Users 2 and 3 (0.707) and Users 4 and 3 (0.729) showed moderate similarity, indicating shared but not identical tastes.

Item-Based Collaborative Filtering: Cosine Similarity: Items such as dumbbells and yoga mats (similarity score of 0.447) were strongly connected, making them ideal for co-recommendation.

Evaluation: Cosine similarity effectively captures similarity in cases where users consistently rate items in similar ways, offering reliable recommendations for users

with highly similar preferences. And Pearson correlation is beneficial when users have varying rating scales, as it accounts for individual biases. It provides a more normalized similarity score, which is useful for detecting less obvious but significant user alignments.

17. Briefly introduce the implementation process, tools and libraries.

Answer: preparing a user-item matrix of ratings. I calculated similarity scores between users or items using both cosine similarity and Pearson correlation. Based on these scores, I calculated predictions by weighting ratings from similar users/items.

Tools and Libraries: Python as a Main programming language for implementation. And NumPy Used for mathematical operations and managing arrays. And Pandas Utilized for data manipulation and handling the user-item matrix.

18. Write your own remarks about the perceived differences between user-based and item-based CF using the similarity measure and the Pearson correlation coefficient.

Answer: User-Based CF Recommends items based on similar users. Cosine similarity identifies users with similar ratings, while Pearson considers both similarity and consistency in rating patterns, giving more balanced results. While Item-Based CF Recommends items based on similar items that users have rated. Cosine similarity here works well for frequently co-rated items, while Pearson helps adjust for different user rating styles.

Cosine Similarity Focuses on exact rating alignment, emphasizing users or items with close matching ratings. While Pearson Correlation Adjusts for individual rating habits, making it better at detecting trends rather than just rating matches.

19. Write a conclusion that demonstrates how each strategy affected predicted accuracy.

Answer: Each strategy impacted prediction accuracy in distinct ways. Cosine similarity, both in user/item-based CF, effectively captured direct preference alignment, leading to accurate predictions when users or items shared strong rating patterns. Pearson correlation, by normalizing user rating scales, adjusted for individual rating biases, resulting in slightly refined predictions. User-based CF with Pearson correlation generally enhanced prediction accuracy for users with varying rating styles.

20. Addresses any enhancement from your point of view.

Answer: we can combine user-based and item-based filtering for better accuracy.

And add user demographics and time factors to personalize recommendations. And we can use ml models like cluster,…

Assignment results:

Average Ratings per User:

1. User1: (5 + 3 + 4+ 4) / 4 = 4.0
2. User2: (4 + 3 + 3 + 2) / 4 = 3.0
3. User3: (4 + 5 + 4 + 3) / 4 = 4.0
4. User4: (2 + 1 + 3 + 4) / 4 = 2.5
5. User5: (3 + 4 + 5) / 3 = 4.0

product_id	treadmill	dumbbells	yoga_mat	elliptical	kettlebell	Cosine(i,3)user	Pearson(i,3) user
User1	5	3	0	4	4	0.993	0.000
User2	0	4	3	3	2	0.995	0.707
User3	4	0	5	4	3	1.00	1.00
User4	0	2	1	3	4	0.804	0.729
User5	3	0	4	0	5	0.940	-0.500
Cosine(2,j)	-1.000	1.000	0.447	-0.333	-0.647		

CF	Prediction for User 3 on Item 2
User-Based CF (Cosine)	3.5
User-Based CF (Pearson)	4.238
Item-Based CF (Cosine)	5 (User 3), 4 (User 5)

Conclusion and Opinion:

Finally, I apply two ways to recommend things: one that focuses on users and another that focuses on items. When I used user-based filtering with cosine similarity, I found that it did a good job of matching people with similar tastes, which helped us predict ratings pretty well. The Pearson correlation also showed some connections between users, giving us a slightly different prediction. However, this method might struggle when there are a lot of users. While item-based filtering showed how similar items could be to each other. For example, dumbbells and yoga mats were seen as closely related. This way of finding recommendations worked

well and was easier to scale up. Simply, user-based filtering was great for finding similar users, while item-based filtering was better at spotting item relationships.

Opinion: I think combining both methods would make the recommendations even better. Using both user and item information, along with smart technologies, could help create a more personalized experience for everyone. And we can use cluster and ml model to improve the similarity and recommender.

Reference:

What is Web Scraping and How to Use It? - GeeksforGeeks

Web Scraping using Python (and Beautiful Soup) | DataCamp

Amazon.com : mini portable treadmills

Matrix Factorization made easy (Recommender Systems) | by Rohan Naidu | Analytics Vidhya | Medium