Images by MidJourney

Product Recommendation Solution

When “**a** to **z**” can be too much!

Recommendation Systems Project
Prepared by Ken Venturi
July 1st, 2023

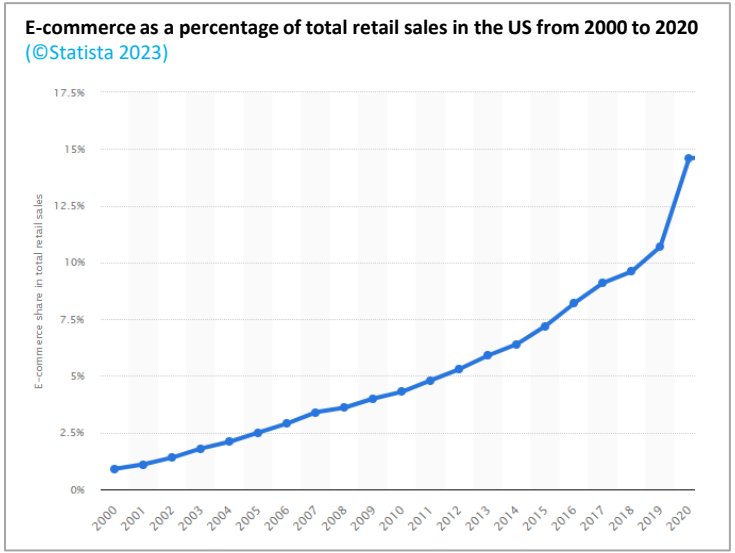
Images by MidJourney

Agenda

- The Problem
- Solution Approach
- Initial Data Exploration
- Model Building
- Conclusion

The Problem - Context

When too much information "***is the***" problem for shoppers.



In the modern era where ecommerce shopping is approaching 20% of all retail sales in the US, a new set of problems has arisen. Of major concern to ecommerce platforms is simply having too much information to process and present to customers.

Too many options often leads to confusion and the "paralysis of analysis" which can negatively affect commerce.

This is where data sciences, machine learning, and recommendation systems come into play. They can assist in providing personalized recommendations that limit the amount of information a user gets to what's most relevant for the user and what's most likely to keep them engaged.

Smart engines like Recommender Systems are used by almost every major E-commerce product around the world to suggest products (movies, songs, clothes, etc.) to their customers.

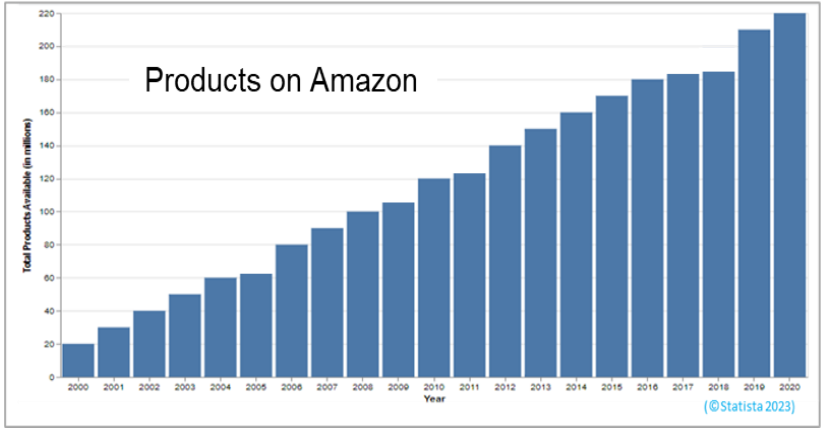
Chart by ChatGPT, ShowMeDiagram, Statisa, 2023

The Problem

Amazon Recommendation

Objectives of the project:

- Extract insights to understand the Amazon review data
- Build a Recommendation System model using different techniques to predict ratings and recommend products
- Compare different models based on various performance metrics and choose the optimal model



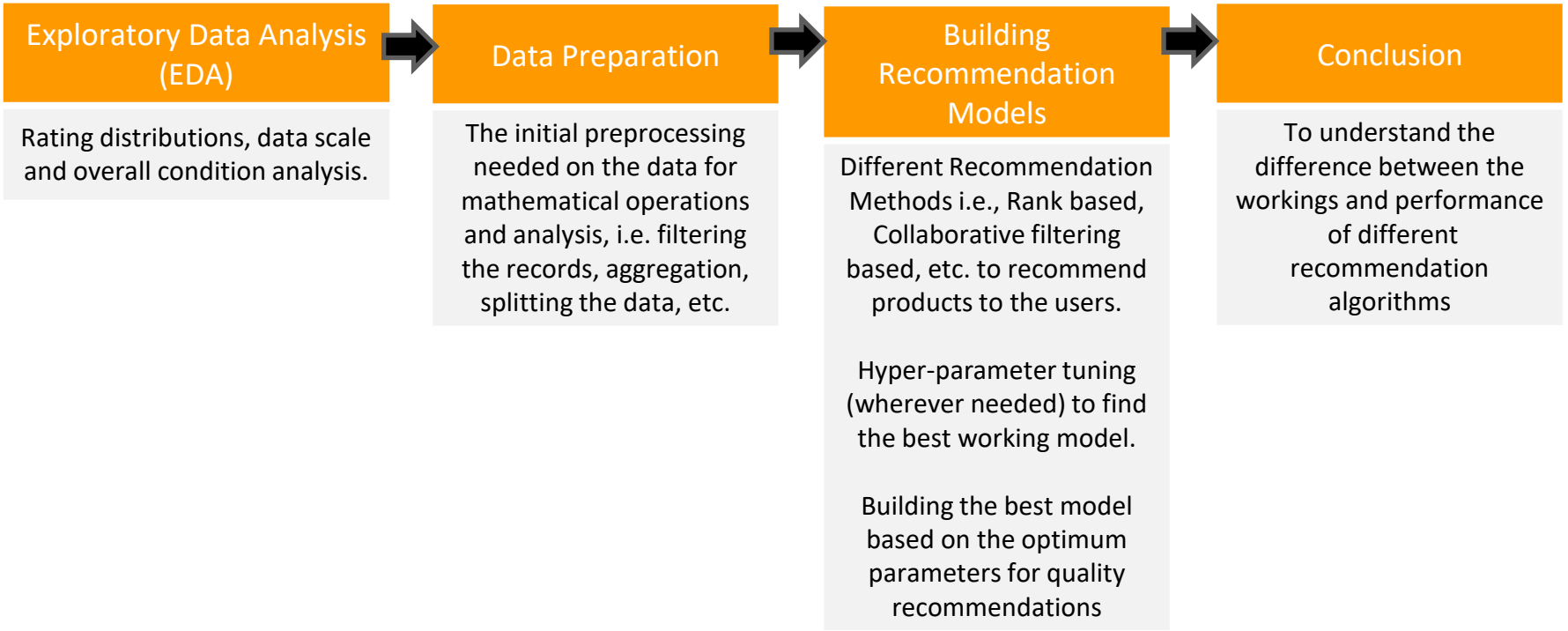
Amazon, the largest of all ecommerce companies, has been challenged by this growth in products on its platform. Amazon serves an important role in recommending various products to users in the US and all around the world. Electronics, clothing, shoes, food, movies, books, music and an endless array of partner products also being sold on the platform are just a few examples.

Amazon collects reviews from users (ratings, text feedback, etc.) and uses that data to recommend various products to them based on their personalized feedback, profiles, and the characteristics of the products themselves.

To enhance customer satisfaction, it is critical to recommend the most relevant items to each user.

Chart by ChatGPT, ShowMeDiagram, Statisa, 2023

Solution Design Approach



Exploratory Data Analysis

(EDA)

Data overview

The Amazon Review dataset contains User ID, Product ID, and Ratings given by the users of the platform. The data does not include text reviews or information, User Names or Product Names. Therefor, results from this analysis will be limited to providing IDs and may therefor require an additional “key” or “join” operator to be employed for useful, final implementation.

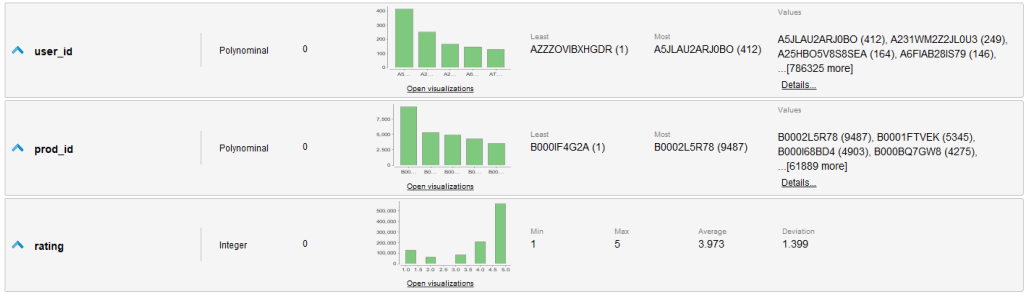
The following are the features and statistics present in the dataset:

User_id	A nominal field that refers to the ID of the user
Product_id	A nominal field that refers to the ID of the Product
Rating	An integer field that refers to the rating a user gives the product upon product review. The scale is 1 – 5 from lowest to highest respectively.

Users	**786,329**
Products	**61,893**
Possible Reviews (All Users X All Products)	**48,668,260,797**
Total Number of Reviews	**1,048,575**
% of Possible	**0 02%**

EDA – initial data review

- Data is in generally good condition for processing
- There are 4 attributes (features)
- No missing values in any of the attributes
- Rating system is on a scale of 1 – 5
- Over 1M records (good scale)
- There are 61,893 unique Product IDs (good sampling)
- There are 786,329 unique User IDs (good sampling)
- Average Rating across all users and all products = 3.973
- With a percentage of possible ratings at 0.02%, we have a relatively small representative dataset of the overall potential user and product interactions

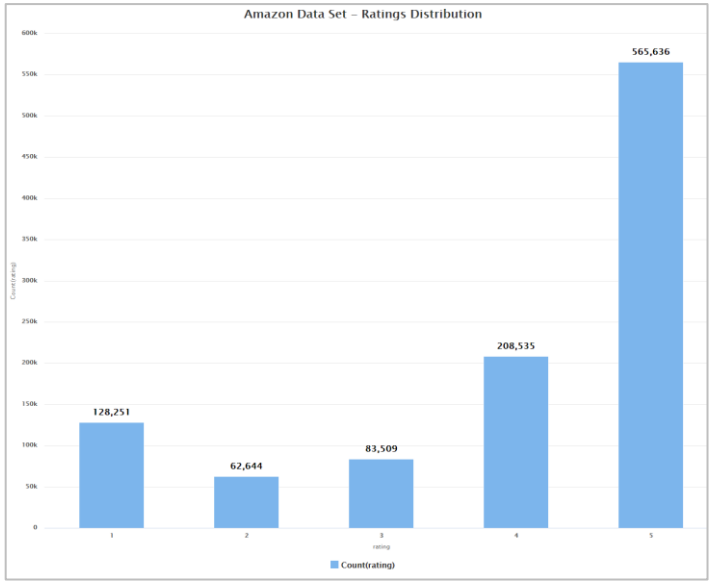


Row No.	user_id	prod_id	rating
1	A2CX7LUOHB2NDG	321732944	5
2	A2NWSAGRHCP8N5	439886341	1
3	A2WNBOD3WNDNKT	439886341	3
4	A1GI0U4ZRJA8WN	439886341	1
5	A1QGNMC6O1VW39	511189877	5
6	A3J3BRHTDRFJ2G	511189877	2
7	A2TY0BTJOTENPG	511189877	5
8	A34ATBPOK6HCHY	511189877	5
9	A89DO69P0XZ27	511189877	5
10	AZYNQZ94U6VDB	511189877	5
11	A1DA3W4GTFXP6O	528881469	5
12	A29LPQQDG7LD5J	528881469	1
13	AO94DHGC771SJ	528881469	5
14	AMO214LNFCEI4	528881469	1
15	A28B1G1MSJ6OO1	528881469	4
16	A3N7T0DY83Y4IG	528881469	3
17	A1H8PY3QHMQQA0	528881469	2
18	A2CPBQ5W4OGBX	528881469	2
19	A265MKAR2WEH3Y	528881469	4
20	A37K02NKUIT68K	528881469	5

ExampleSet (1,048,575 examples, 0 special attributes, 3 regular attributes)

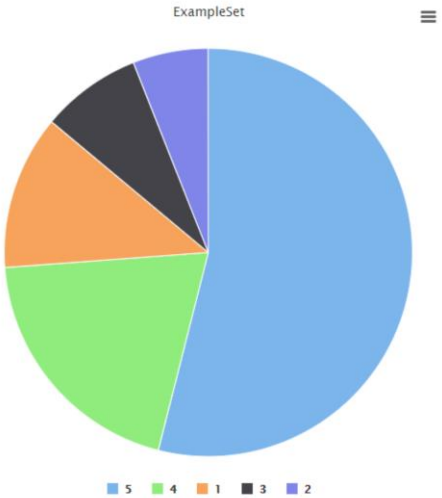
EDA – Distribution of Ratings

Rating Distribution is weighted toward higher ratings with 556K ratings of 5 and over 738k with ratings of 4 or 5



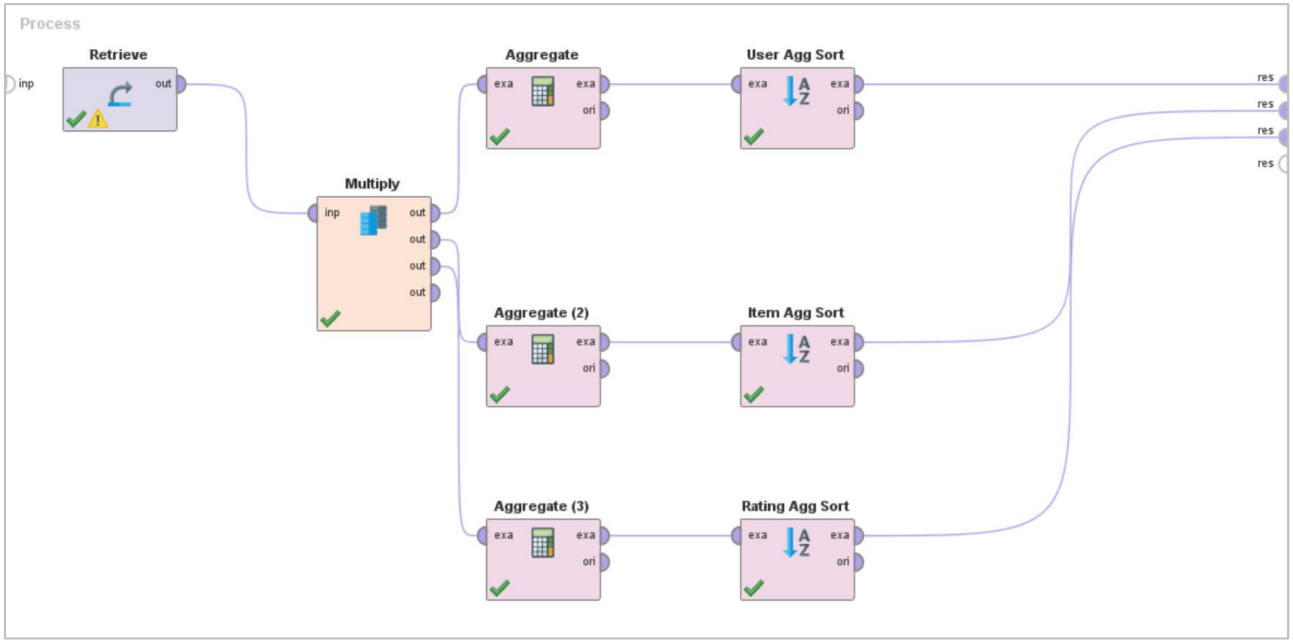
Rating Count

Row No.	rating	count(prod_...
1	5	565636
2	4	208535
3	1	128251
4	3	83509
5	2	62644



EDA – Additional Exploration Model

The model below was developed solely for the purpose of driving additional data exploration on the original dataset.



Top Products and Users (by rating count)

There is a significant ranking variance between the top 10 products and users with reviews, and the bottom rankers with values of just 1 review. Even in the top 10, the top entry is 5X that of the 10th.

Top Products by Rating Count

Row No.	prod_id	average(rati...	count(rating)
1	B0002L5R78	4.449	9487
2	B0001FTVEK	4.007	5345
3	B000I68BD4	3.502	4903
4	B000BQ7GW8	4.553	4275
5	B00007E7JU	4.566	3523
6	B000BKJZ9Q	4.441	3219
7	B000B9RI14	4.776	2996
8	B000A6PPOK	3.950	2828
9	B00007M1TZ	3.977	2608
10	B00004ZCJE	4.124	2547

ExampleSet (61,893 examples, 0 special attributes, 3 regular attributes)

Top Users by Rating Count

Row No.	user_id	average(rati...	count(rating)
1	A5JLAU2ARJ...	3.871	412
2	A231WM2Z2J...	4.309	249
3	A25HBO5V8...	4.963	164
4	A6FIAB28IS79	4.137	146
5	AT6CZDCP4...	3.312	128
6	AKT8TGIT6V...	4.754	122
7	A11D1KHM7...	3.125	112
8	A2B7BUH88...	4.417	103
9	A3OXHLG6DI...	4.421	95
10	A203OCQQ1...	4.433	90

ExampleSet (786,329 examples, 0 special attributes, 3 regular attributes)

Rating Count

Row No.	rating	count(prod_...
1	5	565636
2	4	208535
3	1	128251
4	3	83509
5	2	62644

Data Exploration - Ratings Count of Product

Row No.	prod_id	average(rati...	count(rating)
1	B0002L5R78	4.449	9487
2	B0001FTVEK	4.007	5345
3	B000I68BD4	3.502	4903
4	B000BQ7GW8	4.553	4275
5	B00007E7JU	4.566	3523
6	B000BKJZ9Q	4.441	3219
7	B000B9RI14	4.776	2996
8	B000A6PPOK	3.950	2828
9	B00007M1TZ	3.977	2608
10	B00004ZCJE	4.124	2547

ExampleSet (61,893 examples, 0 special attributes, 3 regular attributes)

- This table shows those product with the highest counts of ratings in the dataset.
- There are a total of 61,893 unique products available in the dataset.
- While this is a very small fraction of the total products available on Amazon.com (220 million plus), it should suffice in designing a model to drive a strong recommendation system.
- While the models being developed in Rapid Miner for this project are suitable to define processes, operators, and data relationships, these “no-code” solutions will likely not suffice as a real time solution with what really is a massive dataset.

Data Exploration - Ratings Count by Users

Row No.	user_id	average(rati...	count(rating)
1	A5JLAU2ARJ...	3.871	412
2	A231WM2Z2J...	4.309	249
3	A25HBO5V8...	4.963	164
4	A6FIAB28IS79	4.137	146
5	AT6CZDCP4...	3.312	128
6	AKT8TGIT6V...	4.754	122
7	A11D1KHM7...	3.125	112
8	A2B7BUH88...	4.417	103
9	A3OXHLG6DI...	4.421	95
10	A203OCQQ1...	4.433	90

ExampleSet (786,329 examples, 0 special attributes, 3 regular attributes)

- This table shows the User IDs for users who have provided the highest number of reviews for product.
- There are a total of 786,329 unique users in the data, and 412 is the highest number of reviews provided by a single user.
- As per the number of unique users and product, there is a possibility of 786,329 x 61,893 = 48,668,260,797 ratings in the dataset. However, we only have 1,048,575 ratings (around .02%), i.e., a very sparse matrix with only a very small fraction of all possible user-product interactions.
- Hence, we can build a recommendation system to suggest any such product a user has not interacted with. For a comprehensive solution that spans the majority of Amazon products a far larger dataset is required and likely different tools to conduct the analytics and ML.

Model Building

Methods Applied

A **Rank-based Recommendation System,** the simplest method of creating recommendation systems, is where we assume that all customers have similar preferences and are seeking information presented in a ranked result. We will employ collaborative filtering systems as our base models for comparison.

User to User

Item to Item

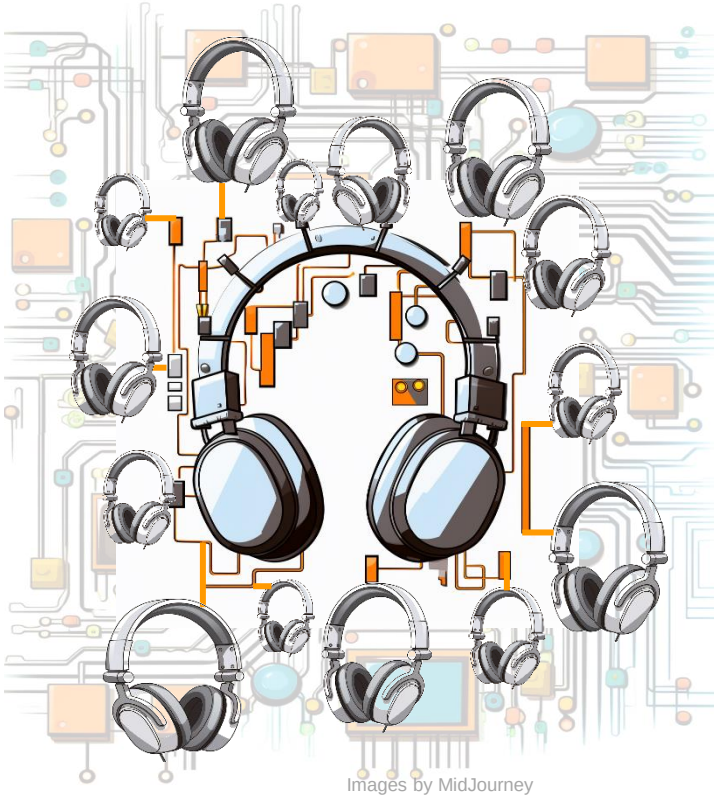
Collaborative Filtering-based Recommendation System

- ***User-User Collaborative Filtering***: Here, an item is recommended to a user based on user-user similarity, by looking at the items used by similar users who have interacted with this item.
- ***Item-Item Collaborative Filtering***: Here, an item is recommended to the user simply based on item-item similarity with items this user has already interacted with.

Images by MidJourney

Performance Metrics

To compare the models, ***RMSE is preferred*** over the other two metrics. The wo primary reasons were that RMSE is used when the data does not contain outliers as all ratings lie between 1 to 5, and more distant examples are penalized in an exponential based formula.



Images by MidJourney

- RMSE: Root Mean Squared Error. This will measure the closeness of predicted ratings to the actual ratings by considering the square root of the sum of squares of the difference between the actual and the predicted ratings. As it considers the square of the differences, it is more sensitive to outliers. The lower the RMSE, the better the model and vice versa.
- MAE: Mean Absolute Error. This is the average absolute difference between the predicted and actual rating given by all the users. The lower the MAE, the better the model.
- NMAE: Normalized Mean Absolute Error. This is a normalized version of MAE in which all error values are normalized to values between 0 and 1. The lower the value of NMAE, the better the model. This metric is used to facilitate the comparison of MAE of datasets with different scales. Not reall relevant to this study but we will look at it anyway.

Model Results: Collaborative Filtering

Here are the performance metrics of different collaborative filtering-based methods applied to the Amazon dataset. You will find three version of each model: 1) Basic Model with setting from project description, 2) Manually Tuned Model, and 3) Optimized Model in which we introduce an optimizer and Cross Validation Operations.

	Most Important			TUNED	TUNED		TUNED	
Model Name	**RMSE**	**MAE**	**NMAE**	**User Min Reviews**	**Data Split (Train/Test)**	**K Value**	**Correlation Mode**	**Time**
User-User Collaborative Filter	1.155	0.892	0.223	100	70/30	90	Cosine	6 sec
User-User Collaborative Filter (tuned)	1.027	0.774	0.193	50	85/15	90	Pearson	8 sec
User-User Optimized (cross validation)	1.643	0.806	0.201	iterated	70/30	iterated	iterated	18 sec
Item-Item Collaborative Filter	1.271	0.981	0.245	100	70/30	90	Cosine	4 sec
Item-Item Collaborative Filter (tuned)	1.289	0.994	0.248	50	85/15	90	Pearson	11 sec
Item-Item Optimized (cross validation)	1.138	0.872	0.218	iterated	70/30	iterated	iterated	80 sec

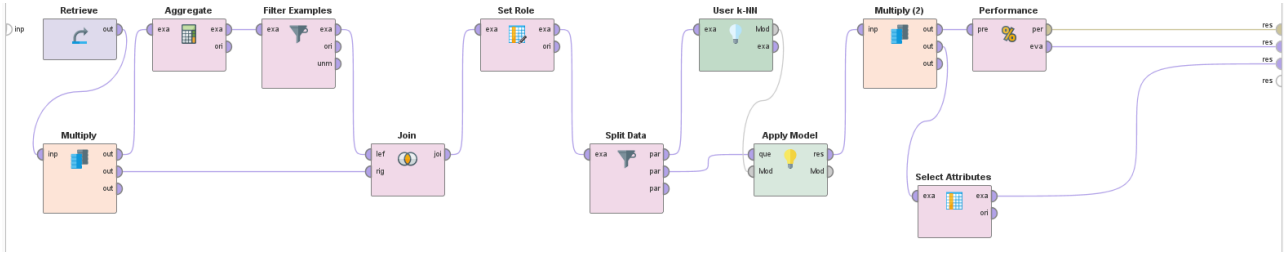
The assignment required the development of the first Basic Model for each collaborative filter method, however, tuning and the use of a Cross Validation optimizer was employed to see if a better performance could be achieved. Clearly that was effective in reducing the overall error rates and ultimately the ***User-User Collaborative Filter (Tuned)*** prevailed.

Chosen “Ratings Prediction” Model

Manually Tuned User k-NN Collaborative Filtering

During the design and testing of the “Optimized Models” we employed an iterative cycle process to test how varying values of key parameters would effect performance. The following were the results of an iterated and then optimized model in the final manually “Tuned Model”.

User to User Rating Prediction



Amazon User-User Colaborative (3 results. Process results)
Completed: Jun 28, 2023 10:06:00 AM (execution time: 6 s)

Performance Vector (Performance)
Result not stored in repository.

```
PerformanceVector:
RMSE: 1.155
MAE: 0.892
NMAE: 0.223
```

Amazon User-User Collaborative (3 results. Process results)
Completed: Jul 2, 2023 10:16:55 AM (execution time: 5 s)

Performance Vector (Performance)
Result not stored in repository.

```
PerformanceVector:
RMSE: 1.028
MAE: 0.774
NMAE: 0.193
```

TUNED

When considering the computational time and resources and the fact that the manually tuned model performed the best of all models, it was clear that the User to User (User k-NN operator) model was the optimal model.

Only a few hyper-parameters appeared to have a real effect on performance results being the ***Correlation Mode***, the number of ***reviews per user filter***, and the ***data split*** enumeration. In this tuned model, those were changed to *Pearson*, *50* and *85/15* respectively.

Results of Chosen model

User to User Collaborative Filtering

Row No.	rating	prod_id	user_id	average(rati...	count(rating)	prediction
1	3	B00004Z0BN	A11D1KHM7...	3.125	112	3.223
2	5	B00005V54U	A11D1KHM7...	3.125	112	3.223
3	4	B000060OEO	A11D1KHM7...	3.125	112	3.223
4	2	B00006483U	A11D1KHM7...	3.125	112	3.223
5	4	B00007DN1E	A11D1KHM7...	3.125	112	3.223
6	2	B00008V6JO	A11D1KHM7...	3.125	112	3.223
7	4	B00008ZPNR	A11D1KHM7...	3.125	112	3.421
8	1	B0000932AM	A11D1KHM7...	3.125	112	3.223
9	2	B000094JJL	A11D1KHM7...	3.125	112	3.223
10	5	B00009W3DS	A11D1KHM7...	3.125	112	3.223
11	3	B0001MKU52	A11D1KHM7...	3.125	112	3.449
12	5	B0006B486K	A11D1KHM7...	3.125	112	3.223
13	3	B000G36GO0	A11D1KHM7...	3.125	112	3.223
14	5	B00004SD8X	A12DLJESJK...	4.385	65	4.291
15	5	B000067VKY	A12DLJESJK...	4.385	65	5
16	5	B00006I5JQ	A12DLJESJK...	4.385	65	4.291
17	5	B00006IRUL	A12DLJESJK...	4.385	65	4.291
18	5	B00007MEHB	A12DLJESJK...	4.385	65	4.291
19	4	B00008NE00	A12DLJESJK...	4.385	65	4.307
20	5	B00008OE5G	A12DLJESJK...	4.385	65	5
21	5	B00008RHA3	A12DLJESJK...	4.385	65	4.307
22	5	B0002SQEWS	A12DLJESJK...	4.385	65	4.291

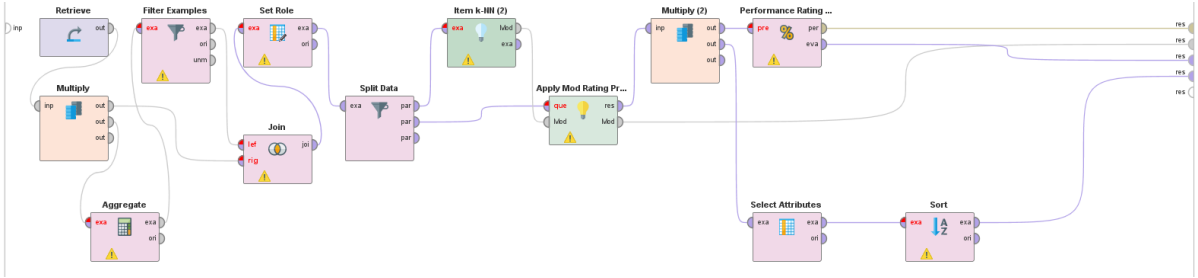
Predicted ratings are fairly reflective of actual ratings in this model's data results. While not perfect, no rating system of this nature is, clear correlations exist in a manner that would have a good success rate across all attempts.

The average MAE (mean absolute error) is 0.774 essentially meaning that in all of the data processed, the average prediction is within 1 star rating of the actual rating for the example.

Model Results of Collaborative Filtering

Below are the base Collaborative Filtering models used in this analysis

Item to Item Rating Prediction



Amazon Item-Item Colaborative (4 results. Process results)
Completed: Jun 28, 2023 10:15:29 AM (execution time: 4 s)

Performance Vector (Performance Rating Prediction)
Result not stored in repository.

```
PerformanceVector:
RMSE: 1.271
MAE: 0.981
NMAE: 0.245
```

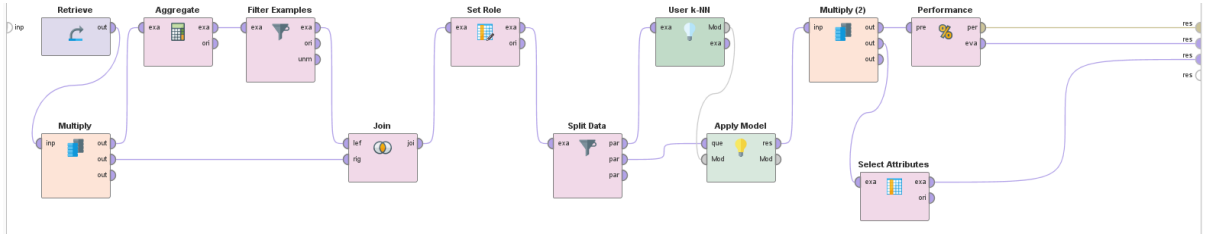
Amazon Item-Item Collaborative (4 results. Process results)
Completed: Jun 29, 2023 11:31:15 AM (execution time: 11 s)

Performance Vector (Performance)
Result not stored in repository.

```
PerformanceVector:
RMSE: 1.289
MAE: 0.994
NMAE: 0.248
```

TUNED

User to User Rating Prediction



Amazon User-User Colaborative (3 results. Process results)
Completed: Jun 28, 2023 10:06:00 AM (execution time: 6 s)

Performance Vector (Performance)
Result not stored in repository.

```
PerformanceVector:
RMSE: 1.155
MAE: 0.892
NMAE: 0.223
```

Amazon User-User Collaborative (3 results. Process results)
Completed: Jul 2, 2023 10:16:55 AM (execution time: 5 s)

Performance Vector (Performance)
Result not stored in repository.

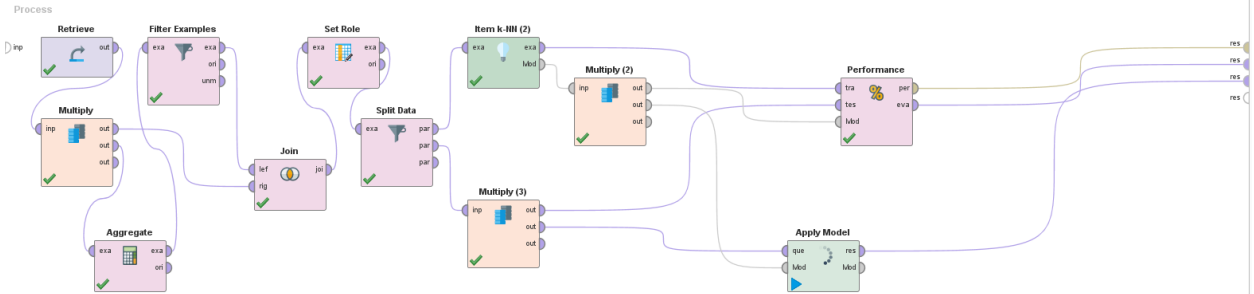
```
PerformanceVector:
RMSE: 1.028
MAE: 0.774
NMAE: 0.193
```

TUNED

Final “Item Recommendation” Model

The manually tuned recommender is what was ultimately chosen. Here the number of reviews per User filter was set to 50 (instead of 100), the correlation mode was set to Pearson and the data split data enumeration was set to 85% Training and 15% Test. This provided the best results and was then employed using the recommender operators to generate the final recommender results.

Amazon Item Recommendation Model (tuned)



Amazon Item Reccomendation Model (3 results. Process results)
Completed: Jun 28, 2023 10:43:04 AM (execution time: 12:11)

Performance Vector (Performance)
Result not stored in repository.

```
PerformanceVector:
AUC: 0.395
prec@5: 0.001
prec@10: 0.001
prec@15: 0.001
NDCG: 0.110
MAP: 0.005
```

Important note on User k-NN Operator in appendix

Final Product Recommendation Model

This table is a sample of the results produced by the "Item Recommendation Model"

	User_ID	Product_ID Recommenation	Rec Rank
1	1309	60505	1
2	1309	6691	2
3	1309	2548	3
4	1309	16436	4
5	1309	32972	5
6	62644	61838	1
7	62644	2692	2
8	62644	8047	3
9	62644	4209	4
10	62644	58119	5
11	425260	61838	1
12	425260	2561	2
13	425260	7851	3
14	425260	8201	4
15	425260	61736	5

- The table shows recommendations of 5 products for each of the three users shown here.
- The user is recommended 5 products with the first being the most preferred item by rank.
- The user_id and product_id are encoded into numbers for ease of data manipulation and will require a "key" or "join" operator to present more useful text names and descriptions to the user.
- While this model confidently produced the best results, application to the full compliment of products that are on amazon.com will require more than a no-code, RapidMiner solution. This does provide a very good illustrative guide to a real-world implementation.

Conclusion

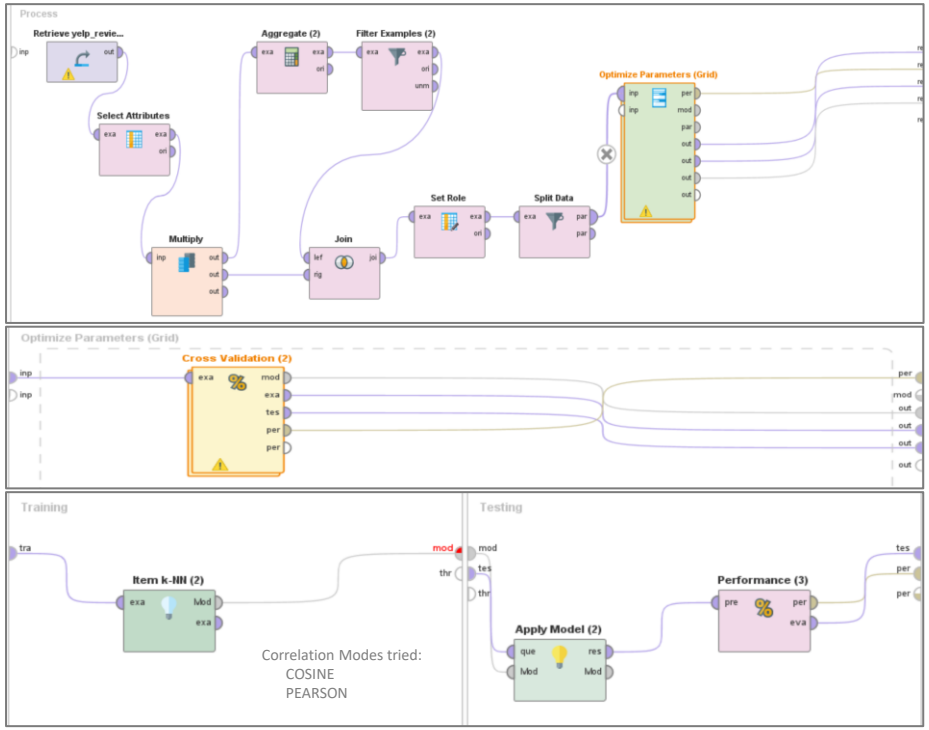
Conclusions

- The vast majority of ratings for product were indeed good or excellent ratings of 4 or 5 out of a 1 to 5 range
- Amazon can improve their Recommendation Systems to suggest relevant product for users, enhance customer satisfaction, and grow the ratings given.
- Out of all possible user-item pairs, in only 0.002% of instances has a user actually rated a product
- The high sparsity of the user-item interaction matrix and the scale of the data means that there is good scope to use a personalized Recommender System to suggest a new product the user has likely not interacted with before.
- It has been observed that the tuned User to User Collaborative Filtering-based model is giving the best performance based on RMSE and the time and resources required to employ the model.
- The company can employ User-User Recommendation models to make personalized product recommendations in order to increase customer satisfaction, engagement and commerce value.

Images by MidJourney

Appendix

Amazon Item Optimized Model – (example purposes only)



These diagrams simply show the full cascading model employed to use a Cross Validation Optimizer.

Ultimately this model was not the best as it was resource intensive and did not provide the best performance results.

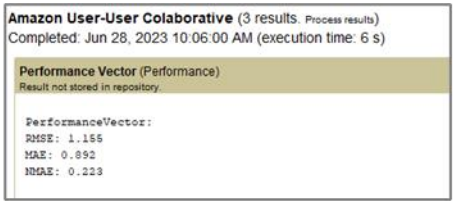
However, the model produced data that did provide many interesting insights as it sorted through the optimizer and cross validation process.

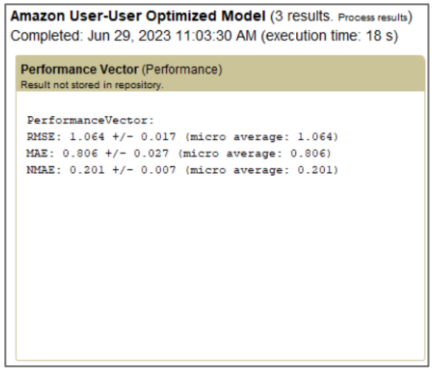
In the end, reviewing the best performing data in this model allowed us to calibrate our Hyper Parameters. That included the ***Correlation Mode*** and ***Filtered Users*** based on number of reviews. This lead to the key parameters added to a different data split enumerations of the final, best performing model.

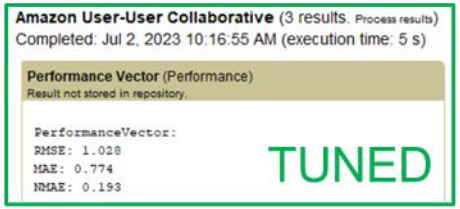
Modifying the hyper-parameter of Reviews per User was a major influencer. It lead to the additional experimentation with removing outliers from the dataset if outliers were defined as “all ratings were 1” or “all ratings were 5” for a user. This would remove the reviewers that only reviewed products in extreme circumstances. After multiple bracketed efforts, we determined this outlier removal tactic did not provide better performance results.

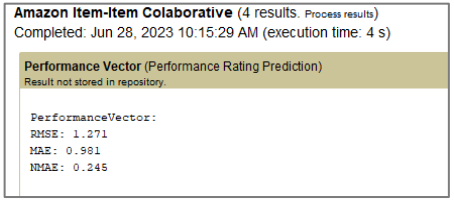
Full Collaborative Filtering comparison chart

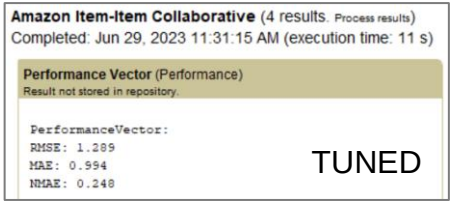
		Most Important			TUNED	TUNED		TUNED	
Model Name	**RMSE**	**MAE**	**NMAE**	**User Min Reviews**	**Data Split (Train/Test)**	**K Value**	**Correlation Mode**	**Time**
User-User Collaborative Filter	1.155	0.892	0.223	100	70/30	90	Cosine	6 sec
User-User Collaborative Filter (tuned)	1.027	0.774	0.193	50	85/15	90	Pearson	8 sec
User-User Optimized (cross validation)	1.643	0.806	0.201	iterated	70/30	iterated	iterated	18 sec
Item-Item Collaborative Filter	1.271	0.981	0.245	100	70/30	90	Cosine	4 sec
Item-Item Collaborative Filter (tuned)	1.289	0.994	0.248	50	85/15	90	Pearson	11 sec
Item-Item Optimized (cross validation)	1.138	0.872	0.218	iterated	70/30	iterated	iterated	80 sec

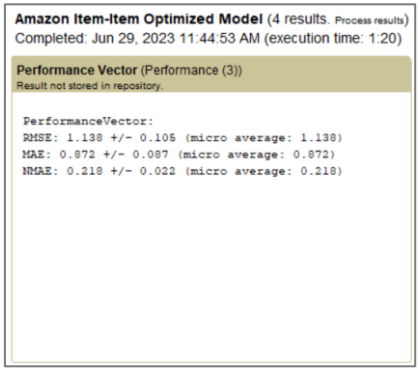












Notes and Future Considerations

User k-NN Operator note on final recommendation model:

After multiple attempts and working on it with other classmates, we determined the “User k-NN operator” (the recommender version) would not operate properly and caused a “Software Bug” error. RapidMiner reported to all of the classmates the same message that for unknown reason, the process is erring out and needs debugging. For that reason, and despite the logic of the performance for the rating prediction models, we employed the Item k-NN operator instead on the final Recommendation Model.

The Need for More Resources and Diffent Tool:

After many many hours of attempting to build a better model, it became apparent that many of these potential processes can be quite computationally demanding. When considering the absolute minute sampling from both the Amazon Product spectrum and the vast number of users, both in the hundreds of millions, the user to item paring of the entire user base across all products becomes problamaticlly huge. Additonal computation resources and developing beyond No-Code tools are likely required to achieve a workflow, dynamic, real time solution.

Image Creation

Charts and graphics were developed using a combination of AI platforms, prompts and plugins as follows:

1. ChatGPT Plus (4.0)
2. ChatGPT Plus (4.0) – PromptPerfect plugin
3. ChatGPT Plus (4.0) – ShowMeDiagrams plugin
4. ChatGPT Plus (4.0) – LinkReader plugin
5. MidJourney – leveraging multiple prompt parameters including: --style, --version, --chaos, and --Nikki

Images by MidJourney

Thank you!

Prepared by Ken Venturi

July 1st, 2023