Amazon Co-purchase Book Recommendation System

Kushal Gandhi[1], Gagan Sankhla[2], Aatman Vaidya[3]

***Abstract*—This is a book recommendation system created based on Amazon co-purchase metadata from SNAP which has different products categorized in Books, music CDs, DVDs and VHS videotapes. By filtering out the data to books, we have applied a book recommendation graph using Social Network Analysis concepts. As a result of this recommendation, we also built a 1-degree ego network graph which gives the idea about neighbouring books of purchased books. We then predict the top 5 books based on SalesRank, AvgRating, TotalReviews, Degree Centrality, and Clustering Coefficient.**

***Keywords*—** *Social Network Analysis, Recommendation System, Amazon Books, SNAP*

I. Introduction

Recommendation System is used to predict future products or preferences on the top of the selected products or priorities. In the past, resources were limited, and users were given limited choices, so there was no need for a recommendation system. The reason behind why there is a need for a recommendation system is that in today's life, there is an abundance of information on the internet, so the user would be confused about choosing the product or item. Many multinational companies provide recommendation services on their online platforms to increase their sales as personalized offers and enhance user experience. The recommendation system increases the speed of searching by providing fast access to the interested content of the user. We aim to apply social network analysis concepts to build and create graphs through this project.

II. Literature Survey

We looked at different existing bodies of work to have a basic understanding of how graph-based recommendation systems would work. (KJ Kim, H Ahn et al., 2012) - Hybrid social network analysis and collaborative filtering approach to enhance the performance of recommender systems. In this research paper, the author collected basic information data from 91 students(gender, age, address) from a university in Korea and the top 100 movie references among them and analyzed their online relationship and the rating of movies. He eliminated one case, which was distorted, so he took 90 respondents, and As an experimental dataset, they used their online relationships and the ratings for 100 movies. (Leskovec, J., Adamic, L. A., & Huberman et al, 2007) - The Dynamics of Viral Marketing, The Program run by large retailers in which the data they collected comes from the user only by giving them some sort of reward. Like discounts and offers on next orders. After that, they analyzed that data which came from 4m people who gave 16m recommendations. The program they want to run behaves like a Real-Life content-based recommendation system. This paper was suggested in the SNAP dataset itself.

III. Dataset

The dataset we have taken is the "Amazon product co-purchasing network metadata" from Stanford Network Analysis Project (SNAP). The dataset is created by taking data from the amazon website and contains product metadata and review information about 548,552 different products. The products are Books, music CDs, DVDs and VHS videotapes. The data was collected in 2006. For each product, the information that is available is the title, sales rank, list of similar products (that get co-purchased with the current product), categories and product reviews.

Dataset statistics	
Products	548,552
Product-Project Edges	1,788,725
Reviews	7,781,990
Product category memberships	2,509,699
Products by product group	
Books	393561
DVDs	19828
Music CDs	103144
Videos	26132

Each product had a certain data format, it is as follows:

1. ID: Product ID (integer)
2. ASIN: Amazon Standard Identification Number
3. Title: Name/title of the product
4. Group: Product group (Book, DVD, Video or Music)
5. Salesrank: Amazon Sales Rank
6. Similar: ASINs of co-purchased products
7. Categories: Location in product category hierarchy to which the product belongs
8. Reviews: Product review information: time, user id, rating, total number of votes on the review, total number of helpfulness votes (how many people found the review to be helpful)

IV. Implementation

1. Data Pre-Processing

Before creating a graph and doing social network analysis, preprocessing is required. The data that we have in

a .txt file is not structured, so our code will not be able to read it correctly. We will have to convert it into a database management system format, from which we will be able to access the data correctly. Similar to a Relational Database Management System (RDBMS). To do so, we kept the ID, ASIN and the Title column as the same. As in all, the incoming data would be stored the same. After that, a new column was created called categories, and this is the transformed version of the categories column from the primary data .txt file. Essentially, in this, the data is cleaned, the ASINs associated with categories are concatenated, some text preprocessing is done: digit, punctuation, stop words are removed, and only unique words are retained. Then after this, we create a co-purchase column where it would be a transformed version of the similar column from the primary data .txt file. All ASINs in a similar field that have been co-purchased are filtered down to only those ASINs with metadata linked with them. After this, the SalesRank, TotalReviews and AvgRating will remain the same. We only want books in our dataset, so we will filter all the books and store that in an empty dictionary.

2. Building the Co-purchase Graph

Now we want to create the graph and further create a recommendation system. We now use the co-purchase data in the dictionary that contains all the filtered book data. The nodes in our graph would be the "ASINs". The graph would be undirected. The graph's edges would be connections between the ASINs (nodes). An edge would exist if two ASINs were co-purchased. Now, each edge would be given an edge weight; the edge weight would be based on similarity, since we wish to make a recommendation system, the similarity would be a solid factor to look at. We would define similarity as

Similarity = (The number of words common in the categories of connected nodes) / (Total number of words in both connected node categories)

The value we get will be between 0 to 1, 1 being the most similar and 0 being the least. The value will be given as the edge weight to each edge. All this data will then be stored in an edge list file; the two columns would be the connected ASINs number, and the third column would have the edge weight of the connected ASIN nodes. To gain a better understanding of the created graph, we also calculate the Degree Centrality and Clustering Coefficient. This would help us understand the proportion of neighbouring nodes connected and grouped.

3. Making Book Recommendations

Now we create a book recommendation system. So, to approach this, we assume that a user has bought a certain book i.e. we take into consideration a certain ASIN number. Here we could just see all the connections that this node has with other nodes, and recommend all the books to the user, but that would not be a good way of going about things. Because, if the Degree Centrality of a certain node is high, then it will recommend all the connected nodes. So thats why, we look at all the metadata associated with that such as, Title, SalesRank, TotalReviews, AvgRating, DegreeCentrality, and ClusteringCoefficient. So that is why, we will now plot ego network graphs for the node book that we will use to recommend other books to the user. An ego graph is a graph that shows the connection of the central node with all its other alter nodes. The central node is the one that we consider as our main node. Here for instance it would be the book that the user has brought.

After plotting the ego graph we filter it to retain the edges that have an edge weight greater than equal to 0.5. This will help us create a better filter for the recommendation system. After this there could be a chance that the degree centrality and clustering coefficient would be high.

Finally we want to recommend the user with 5 top book recommendations, and to even filter out the system, we look at the SalesRank, AvgRating, TotalReviews, Degree Centrality, and Clustering Coefficient. Finally, after this, we recommend the user with top 5 book recommendations.

V. RESULTS

After running the recommendation algorithm on the dataset, the top 5 recommended books with their brief information which were sorted based on the average rating and total reviews. Based on the parched book the ego network depth-1 of that book is displayed along with a 50% threshold ego network graph.

The visualization of our dataset in gephi looks like the following:

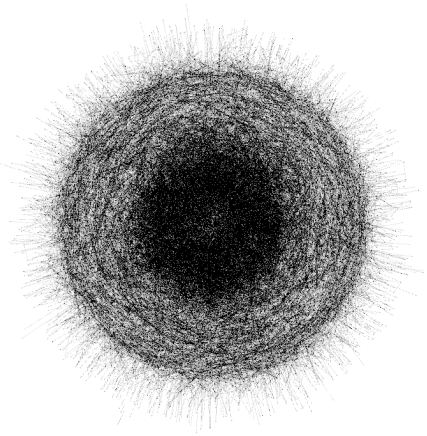

Figure 5.1 - Visualization of the data

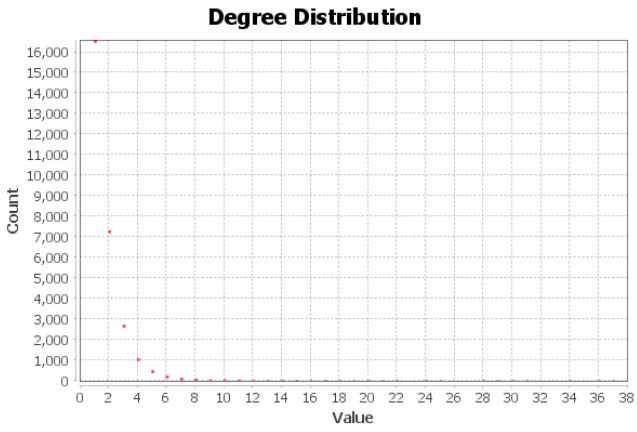

Figure 5.2 - Degree Distribution of the graph

For the result, we have taken an ASIN (node) and we are plotting results for it. The ASIN value taken is *"0764565168"*

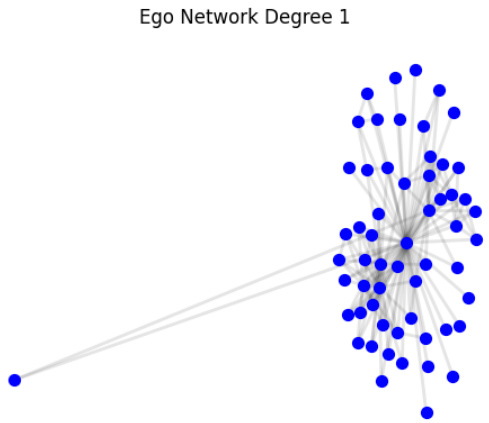

Figure 5.3 - Ego Network of Degree-1

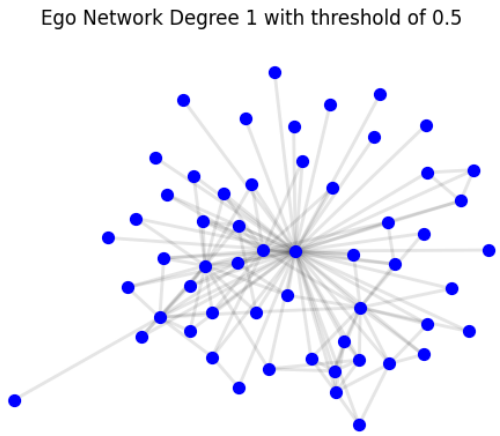

Figure 5.3 - Ego Network of Degree-1 with the threshold of 0.5

The basic information for the ASIN would be outputted as follows.

```
ASIN: 0764565168
--------------------------------------------------------------------------------
-
Purchased Book Info:
--------------------------------------------------------------------------------
-
ASIN:  0764565168
Title:  The Mother of All Pregnancy Books: The Ultimate Guide to Conception, Birth,
and Everything In Between (U.S. Edition)
SalesRank:  658
Total Reviews:  90
Average Rating:  4.5
Degree Centrality:  57
Clustering Coeff:  0.66
```

The top 5 book recommendations would then be suggested to the user.

```
--------------------------------------------------------------------------------
Top 5 Recommendations for the book:
--------------------------------------------------------------------------------
ASIN  Title      SalesRank  TotalReviews    AvgRating  DegreeCentrality
ClusteringCoeff
('0060937645', 'Taking Charge of Your Fertility: The Definitive Guide to Natural
Birth Control, Pregnancy Achievement, and Reproductive Health (Revised Edition)',
265, 701, 5.0, 27, 0.62)
('0060394064', 'Taking Charge of Your Fertility: The Definitive Guide to Natural
Birth Control, Pregnancy Achievement, and Reproductive Health (Revised Edition)',
290410, 701, 5.0, 4, 0.87)
('0060950536', 'Taking Charge of Your Fertility: The Definitive Guide to Natural
Birth Control and Pregnancy Achievement', 8072, 701, 5.0, 4, 0.87)
('1565302656', '1000 Questions About Your Pregnancy', 307833, 30, 5.0, 4, 1.0)
('0789487896', 'Pregnancy and Birth: Your Questions Answered', 169805, 9, 5.0, 2,
1.0)
--------------------------------------------------------------------------------
```

VI. Conclusion

We now have top 5 recommendations of the books by creating a graph between the co-purchased ASINs. This could be extended in the future and a lot more work could be added to it.

1. For instance, we can look at connections between customers and the products that they reviewed, as in people who like X will also like Y.
2. We could create a recommendation network for other groups and products.
3. Link Prediction can be done on the product network and see what clusters form weak and strong link connections.
4. We can also look at connections between groups (products) and customers.

References

[1] Amazon product co-purchasing network metadata. SNAP. (n.d.). Retrieved May 3, 2022, from https://snap.stanford.edu/data/amazon-meta.html

[2] Rodríguez, G. (2018, May 9). Introduction to recommender systems. Tryolabs. Retrieved May 2, 2022, from https://tryolabs.com/blog/introduction-to-recommender-systems

[3] Mall, R. (2019, January 10). Recommender System. Medium. Retrieved May 2, 2022, from https://towardsdatascience.com/recommender-system-a1e4595fc0f0#:~:text=A%20Recommender%20System

%20refers%20to%20a%20system%20that%20is%20capable,to%20the%20prevalence%20of%20Internet

[4] Agrawal-Rohit. (n.d.). Agrawal-Rohit/Amazon-Books-graph-based-recommendation: Amazon Books recommendation using Graph Ego Networks. GitHub. Retrieved April 30, 2022, from https://github.com/agrawal-rohit/amazon-books-graph-based-recommendation

[5] Ben.k. (2018, June 11). Book recommendation (python). Medium. Retrieved April 30, 2022,from https://medium.com/@baemaek/amazon-book-recommendation-system-analysis-d9d72b9a7173

[6] Sivapanuganti. (n.d.). Sivapanuganti/social-network-analysis: Book recommendations using social network analysis. GitHub. Retrieved April 30, 2022, from https://github.com/sivapanuganti/Social-Network-Analysis

[7] Kim, K. J., & Ahn, H. (2012). Hybrid recommender systems using social network analysis. International Journal of Computer and Information Engineering, 6(4), 515-518.

[8] Leskovec, J., Adamic, L. A., & Huberman, B. A. (2007). The dynamics of viral marketing.ACM Transactions on the Web (TWEB), 1(1), 5-es