**Amazon product co–purchasing network:**

**A Recommendation System**

**Rabel Safina**

**Department of Computer Science**

**Sukkur IBA University**

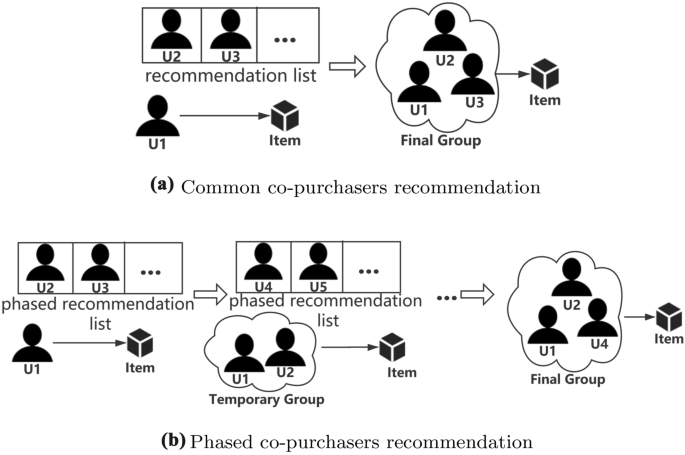
***Abstract-*** **This report describes the recommendation system for amazon products using python language. Eventually, the project focuses itself on to find Customers who bought x also bought y. To recommend a customer to buy a particular product with another product. In this project, I built a product co purchasing recommendation system that makes recommendations given the product name as input. The recommendation based on similar users who bought x and also y. The products the one person has bought that other person have not bought yet can be recommended to him/her. My product co purchasing project work by suggesting products to the person based on metadata. The similarity between the buyers are calculated in last and is used for recommendations.**

***Keywords-* co-products, amazon co-purchasing, recommendation system**

1. INTRODUCTION

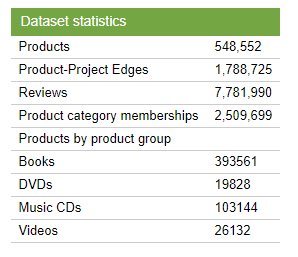
This document is a report of my Social Network Analysis Subject. Teacher give us the project that we have to make some recommendation system or any network based system. Then I chose **Recommendation System for Amazon product co –purchasing network.** It is based on to find Customers who bought x also bought y and to recommend a customer to buy a particular product with another product.





I have made this project using python language on Google Colab. Metadata file-The data was collected by crawling Amazon website and contains product metadata and review information about 548,552 different products (Books, music CDs, DVDs and VHS video tapes). File contains following attributes:

* **Id:** Product id (number 0, ..., 548551)
* **ASIN:** Amazon Standard Identification Number
* **title:** Name/title of the product
* **group:** Product group (Book, DVD, Video or Music)
* **Sales rank:** Amazon Sales rank
* **similar:** ASINs of co-purchased products (people who buy X also buy Y)
* **categories:** Location in product category hierarchy to which the product belongs (separated by |, category id in [])
* **Reviews:** Product review information: time, user id, and rating, total number of votes on the review, total number of helpfulness votes.



1. BACKGROUND

Viral marketing exploits existing social networks by encouraging customers to share product information with their friends. It has been difficult to measure how influential person-to-person recommendations actually are over a wide range of products. Although word-of-mouth can be a powerful factor influencing purchasing decisions, it can be tricky for advertisers to tap into. It is human nature to be more interested in what a friend buys than what an anonymous person buys and to be more likely to trust their opinion and be more influenced by their actions. As one would expect, our friends are also acquainted with our needs and tastes and can make appropriate recommendations.

1. RELATED WORK

Viral marketing can be thought of as a diffusion of information about the product and its adoption over the network. Primarily in social sciences there is a long history of the research on the influence of social networks on innovation and product diffusion. However, such studies have been usually limited to small networks and usually a single product or service. For example, Brown and Reingen [1987] interviewed the families of students being instructed by three piano teachers in order to find out the network of referrals. Similar observations were also made by DeBruyn and Lilien in [2004] in the context of electronic referrals. They found that characteristics of the social tie influenced recipient’s behavior but had different effects at different stages of the decision-making process: tie strength facilitates awareness, perceptual affinity triggers recipient’s interest, and demo-graphic similarity had a negative influence on each stage of the decision-making process.

In the independent cascade model, Goldenberg et al. [2001] simulated the spread of information on an artificially generated network topology that consisted both of strong ties within groups of spatially proximate nodes and weak ties between the groups. They found that weak ties were important to the rate of information diffusion. Centola and Macy [2005] modeled product adoption on small world topologies when a person’s chance of adoption is dependent on having more than one contact who had previously adopted. Wu and Huberman [2004] modeled opinion formation on different network topologies and found that, if highly connected nodes were seeded with a particular opinion, this would proportionally effect the long-term distribution of opinions in the network. Holme and Newman [2006] introduced a model where individuals’ preferences are shaped by their social networks, but their choices of whom to include in their social network are also influenced by their preferences.

1. THE RECOMMENDATION NETWORK

The data was collected by crawling Amazon website and contains product metadata and review information about 548,552 different products (Books, music CDs, DVDs and VHS video tapes).

Problem Statement: Given a query user one has to return the n similar users of that query user and give ranking based on similarity in a user-user network model.

Analyze the Data: The data consists of information about Amazon Products which are of 4 types: Books, DVD’s, Music and Videos. Each product has its description and also consists of reviews given by the user.

Assumptions for the entire problem: User has used the product only if he/she reviews the particular item. As of now the rating of the users is considered uniform: (biasness of the user ratings not considered) Ex: Say for some users (bad-ok-good){0-2,3,4-5} respectively for another user it may be (bad-ok-good){0,2,3}

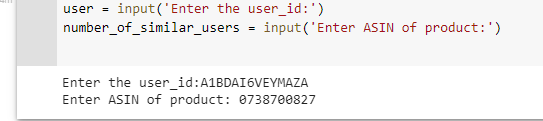
The user- user is linked in a network only if there is at least a single common item that they have rated.(That is two users are directly related if they rate at least one common item)

Data Pre-Processing: Extracted the related information using regular expressions. Created the dictionary of dictionaries for the customer data and item data consisting of different fields as keys and values as the corresponding information.

Building the Model: We build the model such that two users have the direct relationship link if they have rated at least one common item

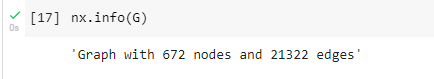
Building the Network: The edge is built between users if they have rated at least one common item. The weight between the user-user is given by summation of decreasing function (rating1-rating2)/ (number\_of\_users\_rated\_for\_that\_common\_item i)/frequency of the items rated by the user. Decreasing function: 5- (rating1-rating2).

Approaches that I used to solve the problem: Used BFS and tried to implement the importance of the flow for a node to calculate similarity. (This is mainly based on the logical approach based on the network/model baseline that I have built). Used N- Random Walks to know the more similar items for that query based on the number of times the node is hit and the weights associated with the neighbors and chances that it goes back to the given query user node given the random walk from a particular query user.

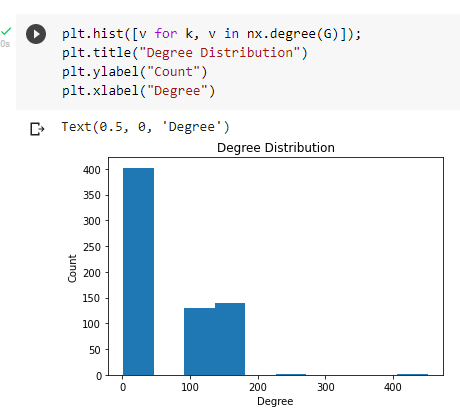
I normalized the weights such that the weights lie between (0-1).Because here we assume that similarity diminishes from a query node as layer increases as we multiply the weights among the different layers. I selected the query node user i and search for the next layer and assign similarity between user I and user j in next layer with edge weights. • For the next layer user k we multiply the before similarity of user j with user k to calculate the similarity between user i and user k If there are more surrounding users from the above layer for node k the average of the weights is taken but ideally it should have extra weightage as many nodes are passing into it so we can attach a constant c based on the number of flow and the layer at which the node is present from the query use. I maintained the dictionary to track the similarity and the similar users and this stops once the criteria is reached that is getting similar users

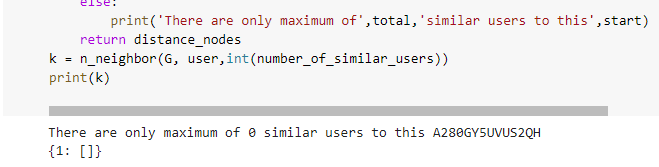
1. RESULTS

I have taken 100 products of Amazon. Plotting their graph:

Displaying graph information:

Entering user id and ASIN Number:

 Degree distribution of products:

In Last finding similar users:

1. CONCLUSION

We saw that the characteristics of product reviews and the effectiveness of recommendations vary by category and price with more successful recommendations made on technical or religious books, which presumably are placed in the social context of a school, workplace, or place of worship. A small fraction of the products accounts for a large proportion of the recommendations. Although not quite as extreme in proportion, the number of successful recommendations also varies widely by product. Still, a sizeable portion of successful recommendations were for a product with only one such sale, hinting at a long-tail phenomenon.

Recommendation algorithms provide an effective form of targeted marketing by creating a personalized shopping experience for each customer. For large retailers like Amazon.com, a good recommendation algorithm is scalable over very large customer bases and product catalogs, requires only sub second processing time to generate online recommendations, is able to react immediately to changes in a user’s data, and makes compelling recommendations for all users regardless of the number of purchases and ratings. Unlike other algorithms, item-to-item collaborative filtering is able to meet this challenge.

1. REFERENCES

<https://www.cs.umd.edu/~samir/498/Amazon-Recommendations.pdf>

<http://www.cs.cmu.edu/~jure/pubs/viral-tweb.pdf>

<http://snap.stanford.edu/data/amazon-meta.html>

<https://colab.research.google.com/drive/18txS40XoH1HkHJQLGMqGtInFAZ57MP-9#scrollTo=xj-0fF7re--p>