Link Prediction in Bipartite Network

Swati Upadhyaya, Sayali Sonawane

Abstract—Most of the real world social networks are inherently bipartite in nature. So, bipartite Networks are one of the most important networks that is used more often to model the complex real world scenarios. It has two different set of elements and a connection exists only between an element in one set to the element in another. One more property of these real world social networks is that they are very dynamic in nature. They update continuously with time. New elements in the network are added, new connections are created among the existing elements or the new elements. One method to study the dynamics of such networks is to predict the new possible connections that might happen over time among the existing elements. In this project, we propose to study the prediction of the missing connections in a social network of bipartite nature.

Index Terms—Bipartite Network, Link Prediction

1 Introduction

A bipartite network is network formed by nodes of two independent disjoint sets and edges existing between the nodes which begin from one of the sets and end in the other. Most of the real world are modeled into a bipartite network to study the relationships among their entities. Some of the examples are Book-Author relationship, Citation Network which consist of relationship between a researcher and his research paper, Actor- Movie relationships, Employee-Company relationships and so on. These networks are also constantly growing networks with new nodes and edges being added to it at every point in time. In this project, we were interested to study where possibly can a new edge be added to the bipartite network with constant number of nodes. This task of estimating a connection between node pairs is called Link Prediction. There are many algorithms available for predicting the missing links in Bipartite Network. But, most of them can be applied on One-Mode Projection of the Bipartite Network. We wanted to study the bipartite networks as a bipartite network rather than on its projection networks. Also, we were keen to know how these algorithms will fare on a social bipartite network with slight modifications. For this purpose, we have chosen the Amazon Product Co-Purchasing Network Metadata available in the Stanford Large Network Dataset Collection as our network data. [4]

Link prediction in unipartite network is usually done with community detection i.e. finding underlying groups or communities in the network. Converting heterogeneous network into homogeneous network. Considering the definition of assortativity as basis, we define communities. Vertices belonging to the same community tend to get connected more often among each other than the vertices belonging to other communities. Such assortative network forms community structure. After identifying this common structure in social network, links are accordingly predicted.

However in bipartite network, the community detection is little different. There are two types of vertices, hence the communities formed in the network are also of two types. Another property of bipartite network is any two vertices of the same community can not connect to each other. Connection is made between the vertices of different types. One way to find community structure in bipartite network is to do the community detection on one mode projection. There are disadvantages while we consider the one mode projection of bipartite network. Information is lost with reduction in dimensionality when only one type of nodes are considered in the network and can lead up to incorrect results. One of the reasons are, when one mode projection is created the customer with low degree in bipartite network has high degree in one mode projection which is misleading. One mode projections are overlap of many cliques which inflates the value of modularity and clustering coefficient. To avoid this issue, community detection on bipartite network is run

The community structure in bipartite network using stochastic block modeling is discussed in [6]. In this project, we try to learn and apply the biSBM algorithm for community detection in bipartite network for link prediction.

1.1 Data Information and Format

This data was collected by crawling Amazon website and contains product metadata and review information for about 0.55 different products. The data is in the following format

1) ID A unique product ID for each product
2) ASIN Amazon Standard Identification Number. Alphanumeric Unique Identifier assigned by amazon.com
3) Title Name of the product
4) Group The product group to which the product belongs. There are three group types Book, DVD, Music or Video

5) Categories A set of one or more the categories belong to. For example, if a product book is titled Resetting the Clock: Five Anti-Aging Hormones That Improve and Extend Life have been categorized as Health and Medicine.
6) Reviews Product review information by the customers who have purchased the products. This review includes product rating, time stamp of the review, total number of votes for the review and total number of helpfulness votes.

Using the above dataset for link prediction in bipartite networks seemed ideal because it could be easily modelled into a Bipartite network of nodes which are customers as one node set and products as another node set and a edge exists between them if a customer has purchased the product. And link prediction in this network would naturally mean recommending a product for the users in the network.

2 ALGORITHM

Our first approach was to use similarity-based algorithms [1,2] to predict the possible connections between the customer and the product nodes that could exist at some point in future. These similarity based approaches assign a score to each non-connected node pair (i,j) in the bipartite network. An edge is expected to exist between the node pair if it has a high score. In our context, it means that, if a link between the non-connected Customer-Product pair has a high value, then it is possible that the customer is likely to buy that product in near future and we recommend that product to him. We compute the scores for every potential node pairs and arrange them in descending order. The top few links in the arrangement which have high scores are highly likely to form in near future i.e. the probability of occurrence of that particular edge is high. We have concentrated on the network topology based measurements. The neighbor based and degree based metrics were used to determine the similarity between the node pairs. In Amazon like social network, people tend to buy products that are similar to the products bought by their neighbors. These measures inherently make use of the proximity of the nodes. Consider node pair (x,y) in the bipartite network where node x is customer entity and node y is a product entity. Now, S(x) is a set of nodes that are two hops away from the node x on the same side as x where x is a customer node. It means that these are the set of customers who buy the same product as customer x. S(y) is the set of nodes which are on the opposite side of node y (one hop away from y), where node y is a product node. So, it is a set of customers who have purchased the product y.

NeighborBased Similarity:

1) Neighbor Based Similarity Common Neighbors (CN) For a given pair of nodes (x,y), common neighbors[2] is a set of nodes that directly have interaction with node x and node y. For a given customer and a given product, common neighbor is the common set of customers that have purchased same products as purchased by node x and they also have purchased product y. So, higher the common neighbor score, higher is the possibility that the customer x will purchase product y in future. It is defined as below

$$CN(x,y) = |S(x) \cap S(y)| \tag{1}$$

2) Jaccard Coefficient (JC) Jaccard Coefficient is the normalized measure of common neighbor metric. It assigns score to the node pair (x,y) , as a fraction of number of common neighbors relative to the total number of neighbors that each of them have. So, it measures the relative degree to which the node pair share common neighbors.

$$JC = \frac{|S(x) \cap S(y)|}{|S(x) \cup S(y)|} \tag{2}$$

3) Adamic Adar Index (AA) - It is a metric that measures the common neighbors features [1,2]. Common neighbors which have fewer features are weighted more heavily. It formalizes the notion that rarer phenomena are more telling. It is defined as below -

$$A(x,y) = \sum_{z \in |S(x) \cap S(y)|} \frac{1}{log(\gamma(z))} \tag{3}$$

where, $\gamma(z)$ is the neighbors of node z .

Node Degree Based Similarity:

- Preferential Attachment (PA) Preferential Attachment [1,2] means that the probability that an existing node acquires a new edge is proportional to its degree. So, it is more likely that a new edge will be formed between the nodes that have high degree values.

$$PA(x,y) = |K(x)| * |K(y)| \tag{4}$$

Method

Fraction of edges were sampled uniformly randomly from the total set of edges considered in the network structure. Using the properties of this set of sampled edges, rest of the edges in the network was estimated.

Evaluation

All the above metrics were evaluated using area under curve. The efficiency of our algorithms was deduced by measuring the area under ROC curve. For each fraction of edges that were presented to the algorithm, we checked how well the algorithm works in predicting the rest of the edges in the network.

Clustering Coefficient of the Bipartite Network

In a unipartite network, clustering coefficient is defined as a fraction of number of observed triangles to the total number of possible triangles. But in a bipartite network, the basic clique is a square. So, clustering coefficient in a bipartite network measures the density if squares in the network. In a bipartite network, it basically calculates the probability that my neighbor have in common except me. It is defined as,

$$C_4(v) = \frac{\sum_{u=1} k_v \sum_{w=u+1} k_v q_v(u,w)}{\sum_{u=1} k_v \sum_{w=u+1} k_v a_v(u,w) + q_v(u,w)} \quad (5)$$

$q_v(u,w)$ are the number of common neighbors of u and w other than v.
$a_v(u,w) = (k_u - (1 + q_v(u,w) + \theta_{uv})(k_w - (1 + q_v(u,w) + \theta_{uw})))$
where θ_{uv} if u and w are connected and 0 otherwise.

To get better understanding of the network structure, we calculated the Clustering Coefficient of the Bipartite Network using the above definition. For the Customer nodes in the Bipartite Network, the value of clustering coefficient turned out to be 0.0028 and for the Product nodes in the Bipartite Network, it had a value of 0.0539. We see that the values of clustering coefficient are very small indeed.

BISBM

Adjacency matrix of bipartite network [6,7] is given as,

$$A = \begin{bmatrix} 0 & B \\ B^T & 0 \end{bmatrix} \quad (6)$$

B is the adjacency matrix of bipartite network considering it as directed network. Considering N_a vertices of type a and N_b vertices of type b. Let there be K_a and K_b groups of type a and type b. B is a N_a x N_b matrix while A is N x N matrix.

Matrix groups are represented in terms of w matrix K x K ($K = K_a + K_b$). A_{ij} = 1 when i and j are connected. w_{rs} = 1 when there is a connected between a vertex in group r with vertex in group s. w_{rs} = 0 when r and s are groups belong to different types of vertices. This equation restricts the network to be bipartite. SBM doesn't assume these restrictions and calculates for all the edges possible. But biSBM calculates only for allowed possible edges which makes biSBM more computationally preferable than SBM.

$$P(G|g,w,T) = \Pi \frac{w_{g_i g_j}^{A_{ij}}}{A_{ij}!} exp^{-w_{g_i g_j}} \quad (7)$$

P is the likelihood of creating a graph G given the parameters g, w and T.
g = groups in both types of vertices
w = adjacency matrix of groups
T = type of the vertices

$$P(G|g,w,T) = \Pi \frac{1}{A_{ij}!} * \Pi_{rs} w_{rs}^{m_{rs}/2} exp^{(} -1/2 n_r n_s w_{rs}) \quad (8)$$

n_r = number of vertices in group r
n_s = number of vertices in group s
m_{rs} = number of edges between groups r and s
Kronecker Delta function is

$$m_{rs} = \sum_{ij} A_{ij} \delta_{g_i,r} \delta_{g_j,s} \quad (9)$$

Taking log on both sides,

$$lnP(G|g,w,T) = \sum_{r,s} m_{rs} ln(w_{rs}) - n_r n_s w_{rs} \quad (10)$$

To maximize the sum, the derivative is set to zero. We get

$$w_{rs} = \frac{m_{rs}}{n_r n_s} \quad (11)$$

After substituting value of w_{rs} into equation of lnP, we get

$$lnP(G|g,w,T) = \sum_{r,s} m_{rs} ln(\frac{m_{rs}}{n_r n_s}) - m_{rs} \quad (12)$$

Second term sums up to 2*number of edges. To maximize the value, we exclude that and just maximize the following equation.

$$P(G|g) = \sum_{r,s} m_{rs} ln(\frac{m_{rs}}{n_r n_s}) \quad (13)$$

To summarize, we maximize this equation to get the best partition possible in the network for bipartite network.

To maximize the above equation over all the groups in g for both types of vertices, we use the algorithm specified in [6]. In the amazon data, the two types of nodes are customers and products. There were 257 different labels. We had to cut it short to 16. We created 32 groups of labels of categories for customers and products. All similar labels were dumped into one label. So products and customers had 16 groups each with same labels but different names for customers and products. Each product with multiple labels on it, belongs to those communities as labels are considered as communities. While customers who purchase that product is tagged with those labels. Hence, there is a community structure in the customers too. We have followed a greedy algorithm here to get the graph with maximum likelihood. Consider all the possible edges in the network and choosing the edge with maximum likelihood. Adding that edge to the network and repeat the process to rest of the edges. The number of edges in the original network should equal to number of edges in the calculated network. The algorithm is computationally heavy as we have to calculate the likelihood for all the possible edges every time after adding an edge to the network.

To predict the links, we calculate this graph with maximum likelihood and check the AUC curve for accuracy. The fraction of edges are observed, and rest of the edges are predicted via the given algorithm.

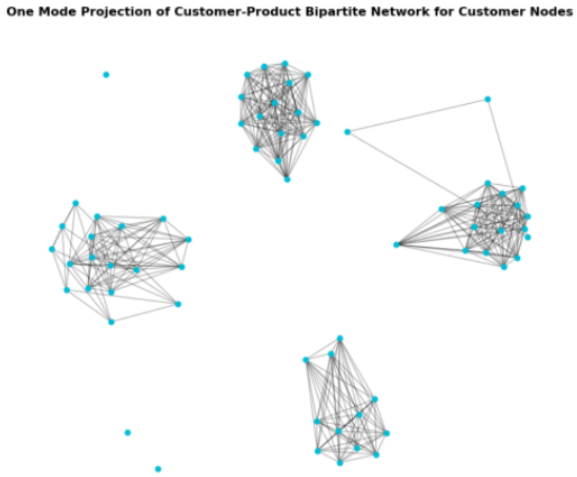

Fig. 1. One Mode Projection. Nodes = 50

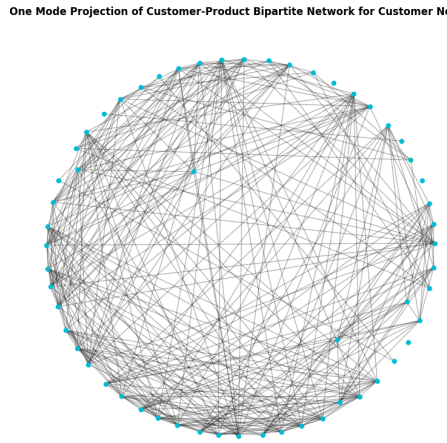

Fig. 2. One Mode Projection. Nodes = 100

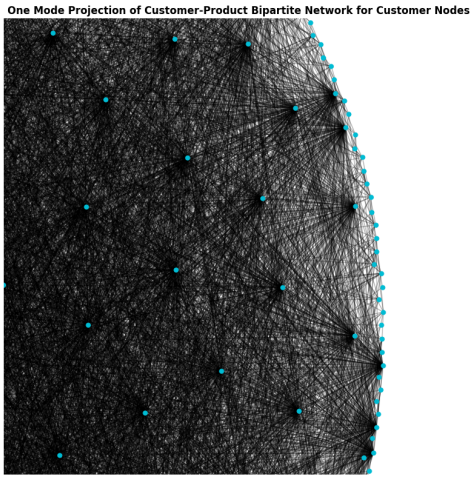

Fig. 3. One Mode Projection. Nodes = 400

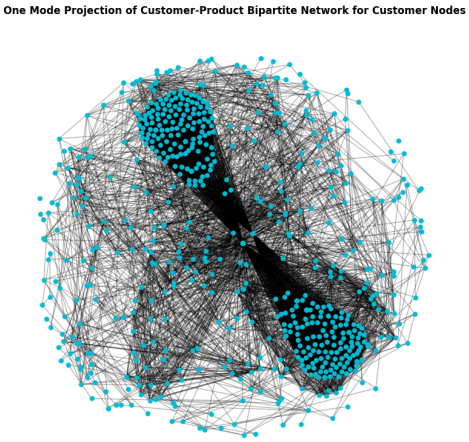

Fig. 4. One Mode Projection. Nodes = 1000

3 Results

4 Discussion

Fig. 1-4 are the snapshot of one mode projection of customer network to show that when the network is small, it shows the highly assortative nature. As the number of nodes increase it changes from highly assortative nature to hairball structure.

Fig. 5 shows the degree distribution of the bipartite network both for customers and products. The maximum customer degree is 387 which means there exists a customer who purchased 387 products. According to the graph, we can see many customers bought less than 10 products. The customer graph follows the power law [8,9] distribution which is true for social networks hence our prediction is right that amazon network is social network which follows social network traits such as assortativity. This just proves

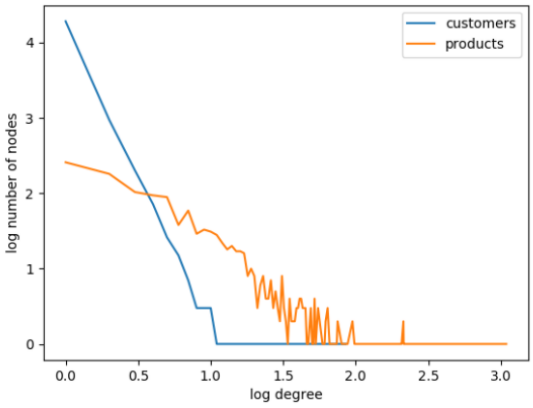

Fig. 5. Degree distribution of bipartite network

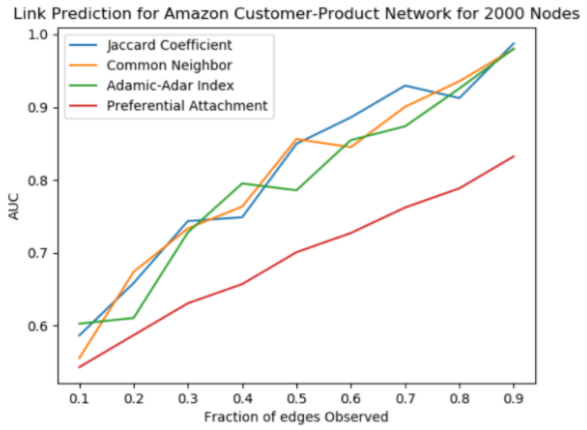

Fig. 6. Accuracy Curve for all the similarity measures

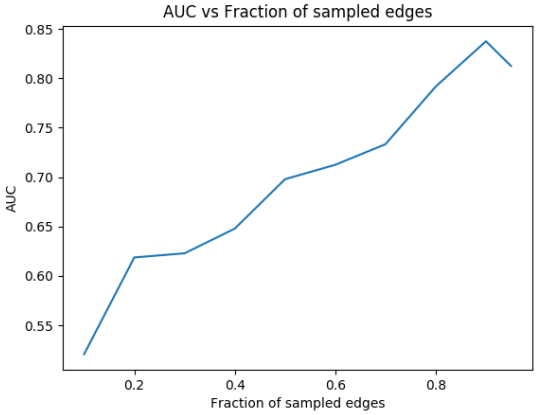

Fig. 7. biSBM Accuracy Curve

that following these algorithms discussed in previous sessions lead to correct link prediction.
Maximum product degree is 2864 which means those many customers bought that product. The degree distribution curve is not a straight line but a curvature.

Fig. 6 shows the accuracy curves for all the similarity measures we discussed i.e. Common Neighbors, Jaccard Coefficient, Adamic-Adar Index and Preferential Attachment. All the three similarity measures Common Neighbors, Jaccard Coefficient, Adamic-Adar Index seem to be geeting the good results since they look at the proximity of the nodes. As per the social tendency goes, customers like to buy products within their communities. While preferential attachment does poorly as compared to other measures because preferential attachment is only looking at the degrees of the customers and products and not their proximity.

Fig. 7 shows the accuracy curve for biSBM model. As it can be seen that biSBM is showing upto 85% accuracy. It is doing poorly due to number of factors. While creating communities out of 257 labels, there might be some human error in their creation. We assumed the network is assortative and focused on that part only while the network has some disassortativity in it, i.e. some noise where there night some edges which are outside of those 2 groups. Finally, we are exploiting the data while using labels. We may have overfitted the data in biSBM. While in other similarity measures we did not use any labels.

5 Challenges

- Unstructured Data: The data was not in form of edgelist but was in a very unstructured format.
- Computationally heavy algorithm: Since biSBM requires many iterations, the time required for it to execute the algorithm on nodes increases exponentially. It takes about more than a day to execute the algorithm.
- Sparse network: Another challenge was the sparse network. From average degree and clustering coefficient, it can be said that the network is sparse. Because of that, the link prediction algorithm does not predict the link correctly.

6 Conclusion / Future work

Conclusion is that most of the customers might have tend to purchase a set of limited products that are extremely popular in nature. They might have had purchased other products based on their obscure interests as well. But the number of people who have purchased this particular set of popular products is high. And these people might not be common neighbors. Such products are not part of many of the closed squares in the numerator but due to their high degree, they will be part of many of the possible squares in the denominator.

Future steps to improve the link prediction could be to exploit the data. But the moment you exploit the data, that algorithm can not be used to other networks. Metadata such as timestamp, rating of the product, the degree of the customer which can be considered as that customer's importance, can be used for better link prediction. Time stamp can be used to get products in trend, rating can be used to get the best products, importance of the customer is used to get the products they purchase and rating they give to the product.

To avoid the heavy computation, we can use AWS or hadoop for parallel computing.

References

[1] Chinta, Kameshwar Kam, Kevin Clark, and Arathi Mani. *quot;Cs224w project final report supervised link prediction in bipartite networks.quot; 2014-11- 13)..* http://snap, stanford, edu/class/cs224w- 2014/projects2014/cs224w-82- final. pdf(2014).
[2] Wang,Peng; Xu,Baowen; Wu,Yurong; Zhou,Xiaoyu. *Link prediction in social networks: the state-of- the-art*. - arXiv preprint arXiv:1411.5118, 2014 - arxiv.org
[3] M Beguerisse Daz. *Analysis of a bipartite network of movie ratings and catalogue network growth models*. - 2008 - eprints.maths.ox.ac.uk
[4] T Zhou, J Ren, M Medo, YC Zhang. *Bipartite network projection and personal recommendation*. - Physical Review E, 2007 - APS
[5] P Zhang, J Wang, X Li, M Li, Z Di, Y Fan. *Clustering coefficient and community structure of bipartite networks*. Physica A: Statistical Mechanics and its Applications ,Volume 387, Issue 27, 1 December 2008, Pages 6869-6875.

[6] Larremore, Daniel B., Aaron Clauset, and Abigail Z. Jacobs. *"Efficiently inferring community structure in bipartite networks."*. Physical Review E 90.1 (2014): 012805. APA
[7] B. Karrer and M.E.J. Newman. *"Stochastic blockmodels and community structure in networks."* Physical Review E 83, 016107 (2011).
[8] M. McPherson, L. Smith-Lovin and J.M. Cook. *"Birds of a Feather: Homophily in Social Networks."* Annual Reviews of Sociology 27, 415-444 (2001).
[9] A. Clauset, M.E.J. Newman and C. Moore. *"Finding community structure in very large networks."* Physical Review E 70, 066111 (2004).