Relationship prediction and link detection of socially connected nodes

Moitrayee Chatterjee
CS Department
Texas Tech University
Lubbock, Texas
Email: moitrayee.chatterjee@ttu.edu

Supra Jyotsna Jampa
Computer Science
Texas Tech University
Lubbock, Texas - 79409
suprajyotsna.jampa@ttu.edu

***Abstract*—In real world netwroks of people, it is observed that we rarely tag our contacts. For e.g. if someone is connected to another 5 people, then it is a rare occassion (s)he mentions who among those 5 connections are his/her "school-friend" or "collegues" or "supervisor" or "family". However, interpreting this type of relationships can be extremely helpful for a number of further applications. For example, if a fund raiser campaign needs to be reached to maximum number of people, and we can detect a 'trust' relationship from the social network of a person, who has donated in that campaign, then the fund raiser can be approached to those connections who has a 'trust' relation with this donor. In this project work, we aim to predict the future relationships between users, who are not yet connected to each other and in our future work we aim to extend our work to infer the relationship types between users.**

Keywords- Link Prediction, Social Ties, Machine Learning, Matrix factorization.

I. Introduction

In real life there are many social networks like Facebook, Twitter, LinkedIn etc. Which are growing every day. All the ties (links) in online networks are very similar to the real life ties but one main difference is the real life ties are labeled (eg. FRIENDS, FAMILLY) while in online networks people do not take time to label them. Many works have focused on mining the relationships in specic domains and one main problem with networks is that there is no labeled data available making it difcult to infer relationships. Also, for protein-protein network or for that matter any real world network. It is useful to predict the future connection between any two nodes of real world networks that would to help in analysis and reconstruction.
In our work, we aim to predict relationships from overlapping human interactions using mathematical method **Matrix Factorization**.

II. Literature review

A. Human Mobility, Social Ties, and Link Prediction [1]

In their work, the authors have considered "Spatio-Temporal Co-occurences" in socially connected users. Their work concentrates on an internet setting and locate those geographic co-events which potentially shapes the derivations about social ties: The fact that two individuals were proximate at only a couple of unmistakable areas at almost similar circumstances can demonstrate a high likelihood that they are socially connected. Utilizing a dataset of 38 million geo-labeled photographs from Flickr, the authors found that the likelihood of a social tie as the number of co-events *(k)* increases sharpely and the transient range *(t)* diminishes. However, the estimations of the probabilities with respect to the baseline likelihood of having a social tie is noteworthy. A probabilistic model is used for learning how movement is related to social ties. Th authors, used a straightforward model initially and then observed the data closely using a richer one. The initial model has *N* isolated geographic cells and *M* individuals, each having one social tie, so that the interpersonal organization comprises of *(M2)* disjoint edges. Every day, each combination of companions visits a place mutually with likelihood β and independently with likelihood $1-\beta$; in either case the decision of location(s) is made arbitrarily. Utilizing Bayes' Law, the likelihood that two individuals are companions (occasion θ) given that they visit the very same cells on k back to back days (occasion D) is:

$$P(\theta|\mathbf{D}) = P(\theta)\frac{P(\mathbf{D}|\theta)}{P(\mathbf{D})} \tag{1}$$

The prior probability that two individuals are companions, P(θ) is $1/(M-1)$, while the likelihood of two connected individuals being at a similar place on a given day is calculated as:

$$p = (\beta) + \frac{1-\beta}{N} \tag{2}$$

This basic probability model captures the macro features but fails to take into account the micro features. Hence the authors takes "homophily" into account. This is the way individuals associated in social ties will probably take part in related exercises, because of their inalienable closeness. The N geographic cells are organized in a framework, and each match of companions (A, B) has a randomized "home" cell, drawn from the 2D experimental dispersion of Flickr photos. Results of this model recommend an approach to evaluate such overlap.

B. Inferring social ties from geographic coincidences [2]

The authors of this paper studied the extent to which the mobility pattern of individual influence the social ties. They used the **C**all **D**etail **R**eport of an anonymous country. The CDR had data for 3 months of activity of 6million users. If

there is a pair of user u and v, then in their work the author adressed three questions: 1. "*How similar is the movement of u and v"*, 2. "*How connected are u and v in the social network", 3*."*How intense is the interaction between u and v*".

Network proximity and Mobile homophily is used in this work for link prediction.

Network proximity: Proximity in mobile network can be used for the future link prediction between two user. From the location obtained from mobile data, if two users are gepgraphically closely located, then there a possibility of them to be connected in near future. The proximity in network is studied using four different measures:

- Common neighbors: Common users who are connected to both *u* and *v*.
- Adamic-Adar: Instead of only counting the common connection, each connected person is given a weigt.
- Jaccards coefficient: It is the size of the intersection of connections of *u* and *v*.
- Katz: Summation over every single conceivable way from *u* to *v* with "exponential damping by length to weight short paths".

Mobile Homophily: In graph based methodologies, predicting new connections by utilizing mobility data can be achieved by searching the degree of closeness in physical space between two people. However, in reality, individuals who share high degree of overlap in their connections, are expected to have a better likelihood of forming new links. The authors used following measures to define similar mobility pattern between individuals:

- Distance.
- Spatial Co-Location Rate. The likelihood of *u* and *v* to visit the same location, but may not be at same time.
- Spatial Cosine Similarity: The cosine similarity of user x and ys trajectories, capturing how similar their visitation frequencies are, assigned by the cosine of the angle between the two vectors of number of visits at each location for x and y.
- Weighted Spatial Cosine Similarity. The tf-idf ver- sion of cosine similarity of the visitation frequencies of users x and y, where the contribution of each location l is inversely proportional to the (log of) its overall population in l.
- Co-Location Rate: Both spatial and temporal vicinity is taken into account and normalized by the number of times *u* and *v* were in same time frame.
- Weighted Co-Location Rate: The likelihood of *u* and *v* to co-exists in same time at same location and it is normalized by the poulation density of that co-location at the given time.
- Extra-role Co-Location Rate: The likelihood of *u* and *v* to co-locate in the same hour of night or over weekends.

The association between individual mobility patterns and social vicinity in the call graph is studied by measuring the connection between *u* and *v* utilizing aforementioned procedures. The quality of the ties in the network is measured by the numbers of calls set between any two individul and also taken into account. Finally, they have formalized the link prediction as a binary classification problem where the network and mobility parameters are used as predictive variables. In short, connections observed in the "past" are used to predict the "future" links.

C. Inferring social ties in Heterogeneous Networks [5]

This paper focuses on a systematic investigation is conducted on the different networks to dene problem and they proposed a Transfer based factor graph (TranFG) model. The model is a framework that considers social theories and applies it to infer social ties in target network.

- Factors influencing the social ties in a network: To infer social ties in a domain specific networks we can devise methods for each networks which cannot be used in heterogeneous networks. Few statistics according to social psychological social theories are used to analyze the network:
  Social Balance: In a network social balance theory can be applied to a triad of nodes, the balanced when there is only one pair of friends or all three of them are friends. Structural Holes: A person in a network is said to span a structural hole if he/she is connected to many other nodes in the network which are not so well connected. This is a very important feature of a node as a node which is a structural hole can be used this node to spread information to its connected nodes. Generally structural nodes are identified for better promotion. eBay is a website which is a structural hole which is connected to many people who may not be well connected to one other. The main is to test if a structural hole tends to have the same type of relationship with the other users. (i)For a node, the total number of pairs of neighbors which are not connected directly are calculated. (ii)The nodes are ranked based on their values and the top 1 percent are considered as structural holes. (iii)If a nodes is a structural hole are the users or most likely 70 percent of the users have same relationship with the node. (iv)Mostly disconnected users have similar type of relationship with the structural hole. Social Status: In this theory we consider directed graphs and each link is given a +/- sign based on where the target node has higher/lower social status than the source node. When we consider a triangle and each negative sign edge is reversed and the sign the resulting triangle is acyclic. Opinion Leader: A opinion leader is node in the network which gets the information first and then passed on to other nodes that are connected to it. These nodes are the ones that are in the higher nodes in a hierarchical networks like networks of employees in a company. (i)The users are categorized into opinion leaders and opinion users by PageRank (ii)The users are ordered and the 1 percent users having highest Page Rank scores are considered as opinion leaders Social theories into semi

supervised learning framework is used to infer the social ties in target network.

- Methodology: Two graphs: Source Network and target network which have few labeled and few unlabeled edges-undirected graph is considered and later a solution for directed graph is proposed. The model is such formulated such that a function f : (GT—GS) → YT
  To infer the relation types in the target network using the supervised information (ie. The labeled of the links in the source network). Input: Two partially labeled networks such that the labels in the source network is much greater than the target network. The type of the relation in the two networks might be very different, even the labels. Since the networks are not completely labeled the predictive function should consider the unlabeled information also. To infer social ties five different networks are considered,
  - Epinions: A network of reviewers where the nodes are users who review products and rate other users reviews with a trust or distrust which form the links.
  - Slashdot: Slashdot is a website where users(nodes) share technology news and can tag other people as friends or foes(links).
  - Mobile: A network of mobile users who are connected with each through calls or texts in the same place. The users are the nodes and there is a link if they communicate with each other.
  - Coauthor: A network from Arnetminer.org which consists of authors and co- authors.
  - Enron: A network of communication in an hierarchical organization of Enron where the employees communicate with each other through emails and the links are of type manager and subordinate.

  Observations when the social theories are applied on the graphs : Domain specific features will change in the case of multiple heterogeneous networks The problem is connected to psychological theories the analysis is conducted on the network based correlations of the following statistics like how social balance and social status are satisfied, the behavior of structural in different networks and the effect of opinion leaders on other networks are used to analyze the network. MODEL:A network is represented as $[G = V, E^L, E^U, X]$
  Where, E^L is the set of labeled edges and E^U is the set of unlabeled edges and $E^L + E^U = E$, the total number of edges. X is the attribute matrix. The formulation is:

$$P(Y|X,G) = \frac{P(X,G|Y)P(Y)}{P(X,G)} \tag{3}$$

When we assume that probability of attributes when the each link's label is conditionally independent, according to Bayes Rule we have

$$P(Y|X,G) = \frac{P(X,G|Y)P(Y)}{P(X,G)} \propto P(X|Y).P(Y|G) \tag{4}$$

To instantiate the probabilities we use Hammersley-Clifford theorem. Using which we get αj = weight of the jth attribute μk = weight of the kth correlation feature function
For the network G with labeled information Y, learning the predictive function is to estimate a parameter configuration,

$$\bigcirc(\theta) = \log P_\theta(Y|X,G) \tag{5}$$

- Model learning and Inferring: To estimate the parameter configuration and to maximize the objective function. A gradient decent method (Newton-Raphson) is used to solve the objective function. An algorithm is deviced based on which the networks are evaluated.
- Algorithm:

Input: a source network G_S, a target network G_T, and the learning rate η
Output: estimated parameters $\theta = (\{\alpha\}, \{\beta\}, \{\mu\})$
Initialize $\theta \leftarrow 0$;
Perform statistics according to social theories;
Construct social theories based features $h_k(Y_c)$;
repeat
  Step 1: Perform LBP to calculate marginal distribution of unknown variables in the source network $P(y_i|x_i, G_S)$;
  Step 2: Perform LBP to calculate marginal distribution of unknown variables in the target network $P(y_i|x_i, G_T)$;
  Step 3: Perform LBP to calculate the marginal distribution of clique c, i.e., $P(y_c|\mathbf{X}_c^S, \mathbf{X}_c^T, G_S, G_T)$;
  Step 4: Calculate the gradient of μ_k according to Eq. 8 (for α_j and β_j with a similar formula);
  Step 5: Update parameter θ with the learning rate η:

$$\theta_{\text{new}} = \theta_{\text{old}} + \eta \cdot \frac{\mathcal{O}(\theta)}{\theta}$$

until *Convergence*;

- Experimental Setup: For the five types of networks, SVM, CRF, PFG, TranFG methods are used for comparison. The experiments conducted for the same features and for different pairs of (source and target) networks, Precision, Recall and F1- Measure are used for evaluation.

Data Set	Method	Prec.	Rec.	F1-score
Epinions (S) to Slashdot (T) (40%)	SVM	0.7157	**0.9733**	0.8249
	CRF	0.8919	0.6710	0.7658
	PFG	0.9300	0.6436	0.7607
	TranFG	**0.9414**	0.9446	**0.9430**
Slashdot (S) to Epinions (T) (40%)	SVM	0.9132	**0.9925**	0.9512
	CRF	0.8923	0.9911	0.9393
	PFG	**0.9954**	0.9787	**0.9870**
	TranFG	**0.9954**	0.9787	**0.9870**
Epinions (S) to Mobile (T) (40%)	SVM	0.8983	0.5955	0.7162
	CRF	0.9455	0.5417	0.6887
	PFG	**1.0000**	0.5924	0.7440
	TranFG	0.8239	**0.8344**	**0.8291**
Slashdot (S) to Mobile (T) (40%)	SVM	0.8983	0.5955	0.7162
	CRF	0.9455	0.5417	0.6887
	PFG	**1.0000**	0.5924	0.7440
	TranFG	0.7258	**0.8599**	**0.7872**

D. Inferring real world connections and discovering, labeling, and characterizing communities[6]

Information has become one of the highly value commodities in the present world. It is used for multiple operations such as optimization of any system, understand the trends and understanding the thought process of people as well. This

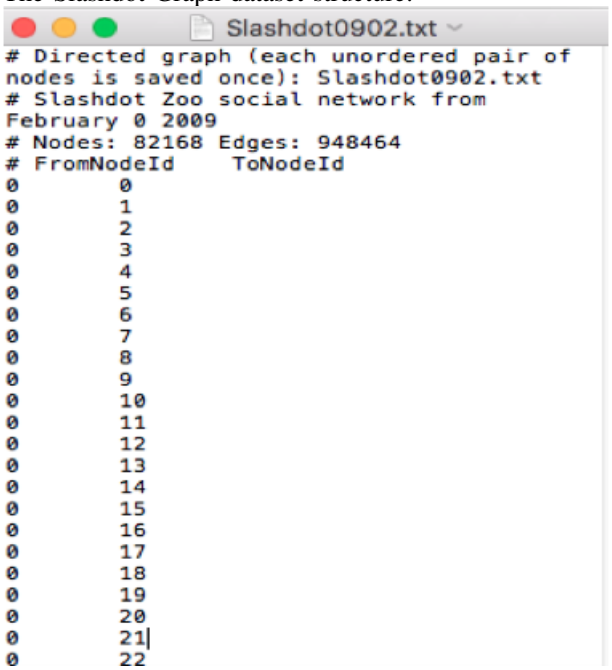
paper tries to analyze the social interactions between people using the information gathered and also the factors which are the underlying reasons for this structure. This is later used to infer the real world connections and the communities of people on the whole. The paper focuses on understanding these social interactions and accordingly the inferences through individual homepages. Individual homepages are something which show us the interests and experiences of human beings. While it is difficult to find individual homepages as they are not free floating on the web, they can be used to infer regarding the communities on the whole. These pages also tend to consist links which consequently link them with people of similar interests and basically form communities of people. This inference can be considered as due to side effect of studying the homepages of people. This is in turn a side effect of people sharing their information on the digital world. In the coming sections, we will discuss about the structure of webpages in terms of small world phenomena, later visualizing the interface for exploring the networks between people. The short world phenomena speaks about how people in the world are connected through a small chain of acquaintances. It basically means that people with similar interests tend to spend time together in smaller circles. Extrapolating from this point of view, it can also be said that people on world wide web and social network on the whole, belong to a small world of its own. In this paper, research has been conducted on homepages of MIT and Stanford students to analyze their networks, interests and accordingly make inference regarding the communities mentioned before. From the study conducted, it was noticed that 70 percent of the MIT users and 30 percent of the Stanford users are connected to other. While people with similar interests might be linked, it is also possible that people with similar links are not connected either. This was successfully eliminated from the scope of study as well. It was also noticed that while users provided out links to two or three, there were users who received many out links. Similarly, there were few users who had multiple links as compared to one or two for most of the users. Clustering coefficient was used to measure the extent to which users band together. It showed that the results for MIT and Stanford was fairly similar whereas the geodesic distance was more for Stanford as compared to MIT. We can infer from these results that MIT and Stanford are part of small world networks where they are closely linked based on the geodesic distance. After creating the networks, in order to understand the reason behind the links, an interface was created which helps them search for the users, notice their details, visualize their network, match the users to others based on their links, text and mailing lists. Once the interface is created, relations can be predicted using a matchmaking algorithm created. For example, likelihood of link between users can be done by finding things in common and also by assigning higher weights to things which are pertained to just certain people.

III. Dataset Description

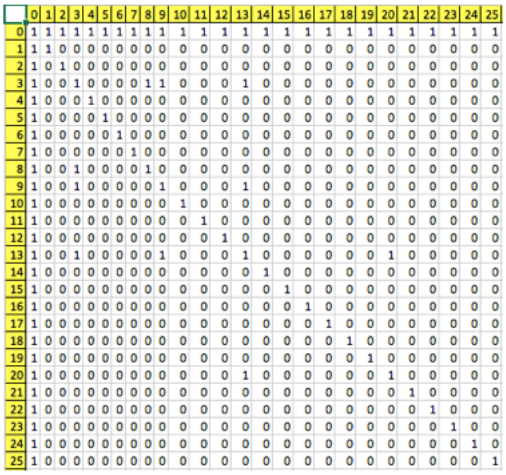
We are using **Slashdot social network, February 2009** [3] dataset. Slashdot is a website for sharing technology related news. On Slashdot one user submit one technology-related news and editor evaluates them. The Slashdot Zoo feature allows the user to tag each other as *friend* or *foe*. In our project we aim to infer ties between users who are not yet connected with each others. The dataset is a directed graph of 82168 nodes representing users and 948464 edges and the following is the snapshot of the dataset:

Name	Type	Nodes	Edges	Description
soc-Slashdot0922	Directed	82,168	948,464	Slashdot social network from February 2009

The Slashdot Graph dataset structure:

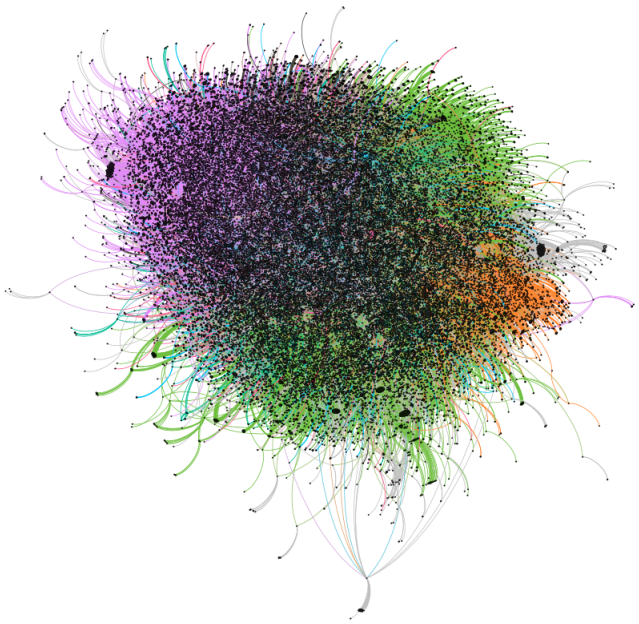


We converted the whole data set into a binary adjacency matrix for using as the input to our code. After coverting the graph dataset to adjacency matrix it looked as below:



Social networks shows a few key charactretistics:
Small world effect: Implies that the average path between any two nodes is small compared to the total size of the network and hence any two nodes can be connected using a short path between them.
The scale free effect: Only a few nodes (hubs or hinge) have lots of connections where as the other nodes are very small in the network.
The clustering effect: It signifies the presence of groups in the network and nodes are connected with each other in a group.

By Gephi visualization we could see that the Slashdot network shows the **clustering effect** as below, where various colors represents different clusters in the dataset:



A single user(node) does not reprsent a whole cluster. and a user, belonging to one cluster can be connected to multiple users from one or more other clusters. With our prediction algorithm, we aim to see all the possible future connections between users.

IV. Technical Approach

We have used a directed, unweighted, graph database which represents the topological structure of Slashdot social network, where there are some existing edges between nodes and our task is to predict if there can be possible links between other nodes, which are not connected in the current dataset. We have used matrix factorization for predicting links. It is an effective mathematical approach that can help in learning the latent features of a tie between two nodes from the adjacency matrix, by calculating the similarity to predict the future link. It is a similarity-based approach where the basic assumption is two nodes which are currently not connected but have a lot more other common connections are likely to be connected in future.

A. Basic Idea

The term "Matrix factorization" refers to the fact of finding two matrices out of one, in a way such that when those two matrices are multiplied, we get back the initial matrix. Matrix factorization can practically be applied to any recommendation system like Netflix or Amazon etc. Let's consider we have a matrix R of P rows representing users and Q columns representing products. The elements R matrix represents the ratings of each product given by each users. Now if we can found two such matrix P and Q where P has k columns and Q has k rows out of a matrix R and compute dot product of P and Q^T then it will give us an pproximation of R'. This forms the model of our work. Now based on the precision-recall (accuracy-fault tolerance) tradeoff we have to provide parameters to our algorithm, such that when we fed the adjacency matrix into the model (or the algorithm) it gave us a new adjacency matrix which is an approximation of the original one. However, in the new matrix some zero-values will be replaced with some new values, signifying possible links. Our Matrix factorization algorithm aims to govern the suitable numbers of latent features of the dataset from the adjacency matrix to rank the nodes based on similarity measures(how similar a node is to another node). This method approximates the likelihood of a future-link between two nodes. The nodes are estimated in a latent space through construction of the adjacency matrix. In social networks, nodes are sparse and it is humanly difficult to calculate the similarity of any two randomly selected nodes without current-connection.

B. Mathematics of Matrix Factorization

We consider our dataset of Slashdot users as a boolean matrix where both columns and rows represents the nodes or users. Each element in the matrix is 1 if two users/nodes have a friendship edge between them and 0 in case of no connection. So, we have to find two matrices **P** and **Q** such that their product will approximate the original matrix **R**.

For example, consider user U1 has rated items I1, I2, I5 as 1,2 and 5 respectively. Another user U2 has rated items I1, I2, I4, I5 as 1,2,5 and 5 respectively. There high chances that user U1 may like product I4 as user U1 and U2 have given similar ratings to the items, this kind of relationship is mathematically formulated to see the correlation between users. The use of matrix factorization, also called collaborative filtering is a very popular method in use for recommendation system, many companies like Netix use matrix factorization to recommend movies to users. Matrix factorization predicts the ratings of items that a user has not yet rated based on the user correlation.

Matrix factorization is used to discover latent features based on which the user rates a product depending the interactions and correlation between the users. Mathematical Formulations:

$$\mathbf{R} \approx \mathbf{P} \times \mathbf{Q}^T = \hat{\mathbf{R}}$$

Here the main aim of matrix factorization is to find two matrices P and Q, such that their product is approximately the original matrix. Relating to the movie recommendation example, if R is the ratings of the items given by users and there are K features associated with the items, Then, P is $(U * K)$ matrix, it represents the association of the users and the features. Q is $(I * K)$ matrix, it is the association of the items and the features.

To predict the rating a user U_i would have given to an item I_i, the dot product of the two vectors are calculated as:

$$\hat{R_{ij}} = p^T q_j = \sum_{k=1}^{k} p_{ik} q_{kj}$$

To find P and Q, one method is to use Gradient descent approach where the two matrixes P and Q are initiated to some values, and their product is calculated and the difference with the main matrix is subtracted from the values and the matrixes are recalculated iteratively till the resultant matrix approximates to the initial matrix. This difference is the squared error between the predicted rating and the actual rating value, the squares of the values are considered as the difference could be either positive or negative and the ratings are positive numbers, the squared values are considered to eliminate negative values. Mathematically:
$e_{ij}^2 = (r_{ij} r_{ij})^2 = (r_{ij} p_{ik} q_{kj})^2$
The difference is used to either increase or decrease the calculated values based on the negative or positive difference. The values are minimized by differentiating the equation and based on the derivatives the p_{ik} and q_{ik} are formulated. α is a constant value which is a rate of approaching the minimum. A very small α is taken because if we consider a large value then the calculated value can be much lesser than the actual values. The method is iteratively applied to minimize the error value.

A regularization parameter β is used to avoid over fitting so that the P and Q are calculated such that the calculation is very close to initial matrix R. The value of β is also small as to properly estimate the values. The functions to calculate the p_{ik} and q_{kj} values are:
$p'_{ik} = p_{ik} + \alpha \frac{\partial}{\partial p_{ik}} e_{ij}^2 = p_{ik} + 2\alpha(e_{ij} q_{kj} - \beta p_{ik})$
$p'_{kj} = p_{kj} + \alpha \frac{\partial}{\partial q_{kj}} e_{ij}^2 = q_{kj} + 2\alpha(e_{ij} p_{ik} - \beta q_{kj})$
The available set of user and item are used to calculate the values of items ratings that are not yet rated.

C. Implementation

We have used R and Python for our implementation. We did our data pre-processing and post-processing using R language. We fed the Slashdot dataset file to our R code and the output was the binary adjacency matrix. The matrix factorization algorithm is implemented in Python. **NumPy** [4] is the basic scientific package that we used in our code and ***xrange(start, stop[, step])*** is the main function we used. The advantage of using *xrange* is it actually *learns* rather than remebering (storing) the results. Hence we get better results. However, depending upon weight and biases this learning rate may vary. At the end of all the iterations of the algorithm a new boolean martix will be obtained and some of the intersection elements containing 0's will now contain 1, signifying possible *friendship*.
Since we can not exactly foresee the future links, to test the our method's precision, a portion of the edges E (0.9 fraction of the entire edges) of some known connection is singled out as a training set ET, the rest of the connections (0.1 fraction of the entire) are utilized as the test set, EP, and no data in EP set is permitted to be utilized for prediction implying, $E = ET \cup EP$ and $ET \cap EP = \phi$.
The prediction quality is assessed by a standard metric, **area under the receiver operating Characteristic curve** (AUC). This metric can be translated as the likelihood that arbitrarily picked missing connection (a connection in EP) is given a higher score than a randomly picked nonexistent connection (a connection in U yet not in E, where U means the widespread set). Among *n* autonomous correlations, if there are *n'* events of missing connections having a higher score and *n"* events of missing connections and nonexistent connection having a similar score, the precision can be defined as:
$AUC = (n' + 0.5n'')/n$

Our algorithm [6]:

```
→ Initialize P and Q with random small numbers
→ for step until max_steps:
    for row, col in R:
      if R[row][col] > 0:
        compute error of element
        compute gradient from error
        update P and Q with new entry

      compute total error
      if error < some threshold:
        break
  return P, Q.T
```

V. Results and Discussions

The matrix factorization is a stochastic gradient decent algorithm and takes a complexity of $\bigcirc(n^3)$. The intuitive explanation can be that there are n^2 latent features needs to be estimated in our dataset. However, this dataset is sparse, hence the performance would be way better than $\bigcirc(n^3)$.
Due to limited computation power of our personal laptops, our algorithm could not be applied to the full dataset. Instead we selected to extract adjacency matrices of 500, 5k and 10K nodes. For the 5k nodes, our algorithm worked considerably well but for 10K datasets we could not finish the exeution without cashing the application. Following figures shows the snapshhot of time taken:

	Data Volume	Time Taken
Training Set	500 * 500	61.11 sec
	5k * 5k	~ 52 minutes

We had to tune the hyper parameters of our algorithm to avoid over-fitting (approximation values too large than the truth) or under-fitting (approximation values too small than the truth) and to get closer to truth results. We have three hyper parameters in our algorithm:

- Learning rate α: Matrix factorization algorithm is a stochastic approximation algorithm and we need to mention the recursive update rate.
- Steps: Number of iterations to get optimal prediction.
- Regularization parameter β: The allowable error limit.

We started our execution with an α of 0.001 and 500 steps and a β of 0.02. But after multiple tunings of the parameters we found an α of 0.002 and 400 steps and β of 0.02 is giving us results which are closer to truth.

After setting the hyper parameter we took the adjacency matrix of our test dataset and randomly changed some 0's to 1's and some 1's to 0's as follow (the cells highlighted in blue signifies the swap in values):

0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
1	0	0	1	0	0	0	0	1	1	0	0	0	0	0	1
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	1	0	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0
1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0

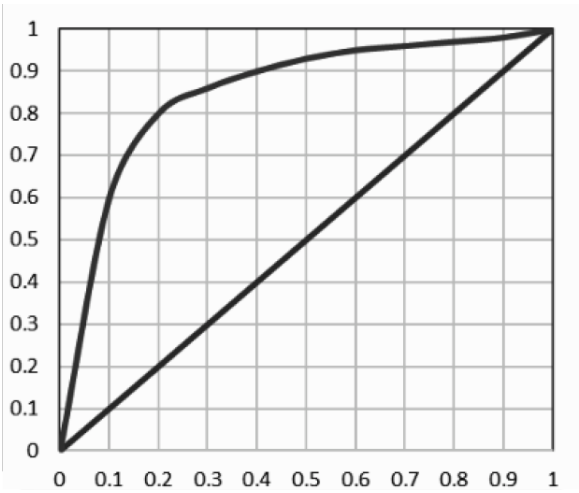
After feeding this twisted matrix to our algorithm the following matrix was received as output:

1.19	1.09	0.93	0.85	0.94	1	1.23	0.89	1.04	1.12	1.02	0.8	1.04	1.09	1.06	1.2
1.1	1.08	0.83	0.81	0.88	0.86	1.14	0.84	1	0.97	1.03	0.68	0.94	0.92	0.95	1.09
1	0.64	0.91	0.62	0.74	1.08	1.02	0.66	0.73	1.16	0.51	0.91	0.95	1.23	1	1.05
0.96	1.01	0.7	0.73	0.78	0.7	1	0.76	0.91	0.8	0.99	0.54	0.81	0.73	0.81	0.95
1.16	1.05	0.92	0.82	0.91	0.99	1.19	0.86	1	1.1	0.97	0.8	1.02	1.07	1.04	1.17
0.92	0.62	0.83	0.58	0.69	0.97	0.94	0.62	0.69	1.05	0.5	0.82	0.87	1.1	0.92	0.97
0.95	0.86	0.74	0.68	0.75	0.8	0.98	0.7	0.83	0.89	0.81	0.64	0.83	0.87	0.84	0.95
0.97	1.03	0.71	0.74	0.79	0.71	1.01	0.77	0.92	0.81	1	0.55	0.82	0.74	0.82	0.96
1.04	0.97	0.82	0.75	0.83	0.87	1.08	0.78	0.92	0.97	0.9	0.7	0.91	0.94	0.93	1.05
1.05	0.85	0.87	0.72	0.81	0.98	1.08	0.75	0.86	1.08	0.76	0.8	0.95	1.08	0.98	1.08
0.84	0.76	0.67	0.6	0.66	0.72	0.87	0.63	0.73	0.81	0.7	0.58	0.74	0.79	0.76	0.85
0.79	0.61	0.68	0.53	0.61	0.77	0.81	0.56	0.63	0.84	0.53	0.64	0.73	0.86	0.76	0.82
0.95	0.76	0.8	0.64	0.73	0.9	0.98	0.67	0.77	0.99	0.67	0.74	0.87	1	0.9	0.98
1.07	1.06	0.81	0.79	0.86	0.84	1.11	0.82	0.98	0.95	1.01	0.66	0.92	0.9	0.93	1.07
1	0.8	0.84	0.67	0.77	0.94	1.02	0.71	0.81	1.04	0.7	0.78	0.91	1.05	0.94	1.03
0.97	0.85	0.78	0.68	0.76	0.85	1	0.71	0.83	0.94	0.78	0.69	0.86	0.93	0.88	0.98

From the above figure we can see that the resulting adjacency matrix from the matrix factorization algorithm is closer to the actual values where a link used to exist but we changed in our dataset, as well as some zero values are updated to signify future links.

The following figure shows the Sensitivity and Specificity trade-off on our test data set. We randomly selected the nodes from the test dataset, especially those nodes, for which we changed the adjacency matrix manually from 1 to 0. We tried to graph plot it with a cut off of 0.5, 0.6 0.7 and 0.75, and assumed a cut off of 0.7. So, any 0's that has been replaced with values greater than or equal to 0.5 after the matrix factorization, signifies a link. Higher the area under curve (AUC), higher the accuracy of the algorithm. The Sensitivity or the True positive rate (TPR) are the values that has been predicted correctly and False positive (FPR) are those wrong results where we changed the 1's to 0's but in resulting matrix the values are less than 0.5. TP rate is calculated as $TP/(TP+FN)$ and FP rate is calculated $FP/(FP+TN)$. These two measures are combined in graph format to signify the accuracy of results.

We plotted Specificity on x-axis and Sensitivity on Y-axis and to get the following ROC plot:



VI. Future work

Due to limited computation power, our current work is not able to handle the total Slashdot datasets. Hence, this work can be extended using a HPC cluster to handle to the full dataset. Besides, the dataset that we used was not labelled. With use of labelled dataset (i.e. having labelled nodes and edges) the algorithm can be modified virtually to any recommendation problem or even to predict the type of relationship between two nodes in any real world network.

VII. GitHub Link to this project

https://github.com/moicha/Inferring-Ties-Between-Disconnected-Nodes

Acknowledgment

We would like to thank our professor Dr. Mahshid R. Naeini for guiding us through out our project.

References

[1] Wang, Dashun, et al. "Human mobility, social ties, and link prediction." Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2011.

[2] Crandall, David J., et al. "Inferring social ties from geographic coincidences." Proceedings of the National Academy of Sciences 107.52 (2010): 22436-22441.

[3] J. Leskovec, K. Lang, A. Dasgupta, M. Mahoney. Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. Internet Mathematics 6(1) 29–123, 2009.
[4] http://www.numpy.org
[5] J. Tang, T.Lou, J. Kleinberg "Inferring Social Ties across Heterogenous Networks".
[6] L. A. Adamic and E. Adar. Friends and neighbors on the web. SOCIAL NETWORKS, 25:211230, 2001.
[7] Websites:
http://www.quuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/
https://www.slideshare.net/BenjaminBengfort/non-negative-matrix-factorization
https://lazyprogrammer.me/tutorial-on-collaborative-filtering-and-matrix-factorization-in-python/
https://arxiv.org/pdf/1503.07475.pdf
http://be.amazd.com/link-prediction/
https://www.wikipedia.org