

# User Recommendation System using Markov Model in Social Networks

Yachana Bhawsar  
Department of Computer  
Application  
MANIT Bhopal

G.S. Thakur, Ph.D  
Department of Computer  
Application  
MANIT Bhopal

R. S. Thakur, Ph.D  
Department of Computer  
Application  
MANIT Bhopal

## ABSTRACT

Today's world is a social world. Recommending resources in social networking is very common thing. There are various methods available to recommend friend, music, video, items in social networks. The users look at the web as a place where they can find an individual or group of people with the same or similar interests, or even find new friends. And many times the recommendation system used in social networks suggests users about these resources. We want to apply Markov models and their variations for addressing this problem. It is generally found that higher order Markov models display high predictive accuracy.

## General Terms

Markov model, Recommendation system, Social network, Transition Probability Matrix

## Keywords

Activity sessions, Predicting links.

## 1. INTRODUCTION

Recommendation system is a subclass of information filtering system. Recommended system is used to predict rating or preference of a user for a given item. There are lots of applications where recommendation system is used such as movies, product etc. In this paper we are talking about social network recommendation system. Social networks recommends about friends, product, page, groups, communities, games, advertisements and organizations. There are three approaches used for recommendation systems: collaboration filtering, content based filtering and hybrid recommender systems.

Collaboration filtering: approaches build a model from a user's past behavior (items previously purchased or selected and/or numerical ratings given to those items) as well as similar decisions made by other users; then use that model to predict items (or ratings for items) that the user may have an interest in [10].

Content-based filtering: approaches utilize a series of discrete characteristics of an item in order to recommend additional items with similar properties [11].

Hybrid recommender system: Recent research has demonstrated that a hybrid approach, combining collaborative filtering and content-based filtering could be more effective in some cases. Hybrid approaches can be implemented in several ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model [12].

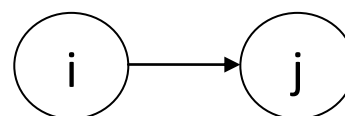
## 2. REPRESENTATION OF SOCIAL NETWORKS

We consider a social network as a undirected graph  $G=(V, E)$  to represent a social network, where edges in  $E$  represents the interaction between  $V$  nodes. If there are two nodes  $u$  and  $v$  in the social network and there is a link between these two nodes or we can say that these nodes are connected with each other. One of the most important elements of social network is edge, and it is interpreted differently in directed and undirected networks. The edge is also called arcs and links. Each edge has some attributes like weight, ranking, type and some properties that depend upon the structure of the rest of the graph. If we talk about weight then it can contain frequency of communication, number of friends and number of communication etc. Ranking includes best friend, second best friend and times of communication etc. Type includes friend, relative, co-worker etc. Properties includes betweenness, similarities etc.

The network data can be represented as adjacency matrix, edge list and adjacency list.

We can understand it with the help of an example, here we consider 13 nodes and there are links between these nodes. After some time some new links are added and shown by dashed lines in the figure 1.

**2.1 Adjacency Matrix:** it represents edges as a matrix and informs that who is adjacent to whom. The entries of the matrix are as follows:



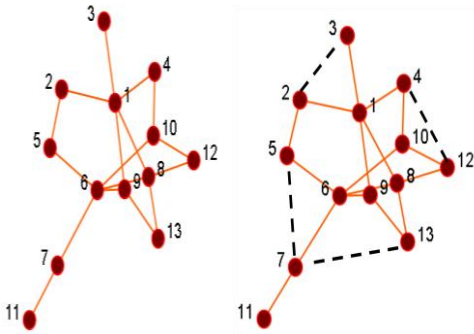
**Fig 1: two nodes of a network**

$C_{ij} = 1$  if node  $j$  has an edge to node  $i$ .

$C_{ij} = 0$  if node  $j$  does not have an edge to node  $i$ .

$C_{ij} = 0$  unless the network has self loop.

$C_{ij} = C_{ji}$  if the network is undirected, or if  $i$  and  $j$  share a reciprocated edge.



**Fig 2: Social network at different time slots**

We are calculating the adjacency matrix for our example network. The table 1 is showing the adjacency matrix . As the network is undirected so the entries are same for i to j or j to i.

**2.2 Edge List:** Edge list is an ordered pair of the nodes, which shows link or edge in directed network. In case of undirected network the node pair is enough. We are showing edge list of our example network. Table 2 is showing the nodes and the corresponding edge list of our example network.

**2.3 Adjacency list:** One of the most important representations is adjacency list. In this type of data representation the node and its adjacent nodes are shown, so we can find the connected nodes. Table 3 is showing the adjacency list of our example network. We can quickly retrieve all neighbors of the network.

**Table 1: Adjacency matrix of example network**

I	1	2	3	4	5	6	7	8	9	10	11	12	13
j													
1	0	1	1	1	0	0	0	1	1	0	0	0	0
2	1	0	0	0	1	0	0	0	0	0	0	0	0
3	1	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	1	0	0	0
5	0	1	0	0	0	1	0	0	0	0	0	0	0
6	0	0	0	0	1	0	1	1	1	1	0	0	0
7	0	0	0	0	0	1	0	0	0	0	1	0	0
8	1	0	0	0	0	1	0	0	0	0	0	1	1
9	1	0	0	0	0	1	0	0	0	0	0	0	1
10	0	0	0	1	0	1	0	0	0	0	0	1	0
11	0	0	0	0	0	0	1	0	0	0	0	0	0
12	0	0	0	0	0	0	0	1	0	1	0	0	0
13	0	0	0	0	0	0	0	1	1	0	0	0	0

**Table 2: Nodes and edge list**

Node1	Node2	Edge list
1	2	(1,2)
1	3	(1,3)
1	4	(1,4)
1	8	(1,8)
1	9	(1,9)
2	5	(2,5)
4	10	(4,10)
5	6	(5,6)
6	7	(6,7)
6	8	(6,8)
6	9	(6,9)
	10	(6,10)
7	11	(7,11)
8	12	(8,12)
8	13	(8,13)
9	13	(9,13)
10	12	(10,12)

### 3. RECOMMENDATION SYSTEM DEFINITIONS AND STRUCTURE

A recommendation system can be represented as a tuple. The definition is as follows:

**Definition 1:** a recommendation system is a tuple  $RS = (U,R,A,Y)$  where U, R, and A are finite sets with elements users, recommended item, edge attributes respectively, while Y is a relation between the sets i.e., Y is subset of  $U \times T \times R$ . The relation  $(uk, ri, ak)$  means that the user  $uk$  recommended for item  $ri$  to edge attribute  $ak$ .

The processes of recommending a friend relay on three different approaches to determine similarity between users (based on [13]): link matching—analysis of graphs created based on social networks, content matching—similarities of contents posted by the users, and content and link matching—the content matching algorithm enhanced with social links and information derived from the structure of social networks.

**Table 3: Adjacency list**

Node	Neighbour Nodes
1	2,3,4,9,8
2	1,5
3	1
4	1,10
5	2,6
6	5,7,8,9
7	6,11
8	1,6,12,13
9	1,6,13
10	4,6,12
11	7
12	8,10
13	8,9

#### 4. DATA REPRESENTATION OF USER INTERESTS AND ACTIVITIES

Social networks users constantly interact with their peers, add new items and use different edge attributes. Social networking sites recommended different items to users. All activities of users can be represented as a 3-D matrix with the following dimensions: users, resources/items recommended, and edge attributes. If a user  $u_m$  used a resource  $r_n$  to edge attributes  $a_k$ , then a point in the 3-D matrix  $(U, R, A)$  with coordinates  $(m, n, k)$  is marked. Therefore, if we want to look at activities of a single user we take a single sheet of the  $(U, R, A)$  matrix that is related to a given user  $u_m$ , i.e., we take a matrix  $(u_m, R, A)$ . An example of a matrix  $(R, A)$  for the user  $u_m$  is presented in Table 4. Here we are taking 13 different users and 14 different edge attributes. The table 4 showing marks for the user with corresponding edge attributes.

#### 5. MARKOV MODELS FOR WEB PAGE PREDICTION

Markov model for web page prediction and user behavior prediction is used in many researches. Markov models are represented by three parameters  $\langle A, S, T \rangle$ , where  $A$  is the set of all possible actions that can be performed by the user;  $S$  is the set of all possible states for which the Markov model is build; and  $T$  is a  $|S| \times |A|$  Transitivity Probability Matrix (TPM), where each entry  $t_{ij}$  corresponds to the probability of performing the action  $j$  when the process is in state  $i$ .

The input data for building Markov models consists of sessions where each session consists of number of pages. In this problem, the actions for the Markov model correspond to the different pages in the web, and the states correspond to all consecutive pages of length  $K$  that were observed in the different sessions. In first-order models, the states will correspond to single page, in the case of second order models,

the states correspond to all pairs of consecutive pages, and so on.

**Table 4 :Slice of  $(u, r, a)$  matrix representing activities of a single user**

User S	U	U	U	U	U	U	U	U	U	U	U	U	U
	1	2	3	4	5	6	7	8	9	10	11	12	13
Edge attri butes													
A <sub>1</sub>	√	√							√	√		√	
A <sub>2</sub>			√				√	√			√		√
A <sub>3</sub>				√	√	√							
A <sub>4</sub>	√	√		√	√		√		√	√		√	√
A <sub>5</sub>			√			√		√			√		
A <sub>6</sub>		√		√	√		√		√	√		√	√
A <sub>7</sub>	√		√			√		√			√		
A <sub>8</sub>		√			√		√		√			√	
A <sub>9</sub>			√			√							
A <sub>10</sub>	√	√		√			√		√		√		√
A <sub>11</sub>					√								
A <sub>12</sub>		√	√			√				√			
A <sub>13</sub>				√				√				√	
A <sub>14</sub>	√			√							√		

The transition probability matrix is calculated once the states of the Markov model are identified. There are many ways for it. The most commonly used approach is to use a training set of action-sequences and estimate each  $t_{ji}$  entry based on the frequency of the event that action  $a_i$  follows the state  $s_j$ .

Web page prediction using markov model is done as fallows. Here we are taking different time slots as a web session(WS) and sequence of pages. We can uderstand it with an example. We consider 5 different web sessions and then we calculate transition probability matrix for first order and second order markov model.

WS1: {P3, P2, P1}

WS2: {P3, P5, P2, P1, P4}

WS3: {P4, P5, P2, P1, P5, P4}

WS4: {P3, P4, P5, P2, P1}

WS5: {P1, P4, P2, P5, P4}

**Table 5: 1st order Transition Probability Matrix**

1 <sup>st</sup> order	P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	P <sub>4</sub>	P <sub>5</sub>
s <sub>1</sub> ={P <sub>1</sub> }	0	0	0	2	1
s <sub>2</sub> ={P <sub>2</sub> }	4	0	0	0	1
s <sub>3</sub> ={P <sub>3</sub> }	0	1	0	1	1
s <sub>4</sub> ={P <sub>4</sub> }	0	1	0	0	2
s <sub>5</sub> ={P <sub>5</sub> }	0	3	0	2	0

To construct 1st order transition matrix we take the example of third web session WS3. Each state is made up of a single page in first order Markov model. So the first page P4 corresponds to the state s4. Since page P5 follows page P4 the entry t54 is updated in the TPM. Similarly, the next page P2 and the entry t42 will be updated in TPM. In the higher order Markov model the states have more than one actions, so for a second-order model the first state of web session will consist of pages {P4,P5} and page P2 will be updated. Once the TPM is built then prediction of link is easy, we can see the highest score in TPM is the predicted link. In our example for WS3 it is P5 if we are taking 1st order Markov model.

**Table 6: 2nd order Transition Probability Matrix**

2 <sup>nd</sup> order	P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	P <sub>4</sub>	P <sub>5</sub>
{P <sub>1</sub> ,P <sub>4</sub> }	0	1	0	0	0
{P <sub>1</sub> ,P <sub>5</sub> }	0	0	0	1	0
{P <sub>2</sub> ,P <sub>1</sub> }	0	0	0	1	1
{P <sub>2</sub> ,P <sub>5</sub> }	0	0	0	1	0
{P <sub>3</sub> ,P <sub>2</sub> }	1	0	0	0	0
{P <sub>3</sub> ,P <sub>4</sub> }	0	0	0	0	1
{P <sub>3</sub> ,P <sub>5</sub> }	0	1	0	0	0
{P <sub>4</sub> ,P <sub>2</sub> }	0	0	0	0	1
{P <sub>4</sub> ,P <sub>5</sub> }	0	2	0	0	0
{P <sub>5</sub> ,P <sub>2</sub> }	3	0	0	0	0

## 6. PROPOSED METHOD

We are proposing Markov model for user recommendation in social networks. We are giving an algorithm for the proposed work. The algorithm steps are follows:

Input: a social network.

Output: recommendation of next activity.

Step 1: Identify number of users and number of activities

Step 2: Construct matrix of users and activities.

Step 3: Identify sessions.

Step 4: Construct transition probability matrix for different order Markov model.

Step 5: Evaluate result.

## 7. ILLUSTRATIVE EXAMPLE

Here we are taking 14 different edge attributes, also called features. We are taking data which contain links as user id of two friends, and different features. Initially all features contain value zero, we update the data when the user perform the corresponding activity. The next activities which user performs contain the higher number. So the last activity performed by user is the highest number among the activities. Table 7 is showing the users and activities.

Next work we have to do is making sessions of activities, so that we can predict the next activity performed by the user. For example if we are taking user with user id U1. So the session is {A4, A2, A7, A10, A14}. We can make all activity sessions like this only.

Activity sessions:

U1 :{ A4, A2, A7, A10, A14}

U2 : { A6, A4, A1, A10, A8, A12}

U3 : { A5, A9, A7, A12 }

U4 : { A10, A13, A14, A3, A4, A6}

U5 :{ A4, A3, A8, A6, A11}

U6 :{ A3, A7, A12, A9, A5}

U7 :{ A8, A10, A6, A2, A4}

U8 :{ A7, A2, A13, A5}

U9 :{ A6, A1, A4, A8, A10}

U10 :{ A1, A4, A12, A6}

U11 :{ A7, A2, A5, A14, A10}

U12 :{ A1, A6, A4, A13, A8}

U13 :{ A2, A6, A10, A4}

Now we can construct transition probability matrix of first order, second order and higher order models. The complexity of constructing transition probability matrix is increases as the order of the Markov model is increases. So we are showing only 1st and 2nd order transition probability matrix.

By observing 1st order transitivity matrix we can recommend activities to users. For example if we want to recommend user U1 then we can see the last activity in the U1's session is A14. By looking transition probability matrix we can say that U1 user is recommended for activity A3 or A10. In this way we can say that user U2 is recommended for activity A6 or A9. There is a strong recommendation to user U4 to the activity A4 because the last activity performed by user U4 is A6 and we can see in transition probability matrix that A6 has highest frequency for activity A4. Now we are constructing 2nd order transition probability matrix for better recommendation. Table 9 is showing 2nd order transition probability matrix.

The 2nd order transition probability matrix is giving 13 different states. The state space is increases as the order of the Markov model increases. Here we have patterns of activities and the prediction accuracy is improves as the order of model increases. We recommend the next activity to a particular user by observing last 2 activities of that user.

**Table 7: Users and activities**

Users ↓ Activities	U <sub>1</sub>	U <sub>2</sub>	U <sub>3</sub>	U <sub>4</sub>	U <sub>5</sub>	U <sub>6</sub>	U <sub>7</sub>	U <sub>8</sub>	U <sub>9</sub>	U <sub>10</sub>	U <sub>11</sub>	U <sub>12</sub>	U <sub>13</sub>
A <sub>1</sub>		3							2	1		1	
A <sub>2</sub>	2						4	2			2		1
A <sub>3</sub>				4	2	1							
A <sub>4</sub>	1	2		5	1		5		3	2		3	4
A <sub>5</sub>			1			5		4			3		
A <sub>6</sub>		1		6	4		3		1	4		2	2
A <sub>7</sub>	3		3			2		1			1		
A <sub>8</sub>		5			3		1		4			5	
A <sub>9</sub>			2			4							
A <sub>10</sub>	4	4		1			2		5		5		3
A <sub>11</sub>					5								
A <sub>12</sub>		6	4			3				3			
A <sub>13</sub>				2				3				4	
A <sub>14</sub>	5			3							4		

**Table 8: 1<sup>st</sup> order transition probability matrix**

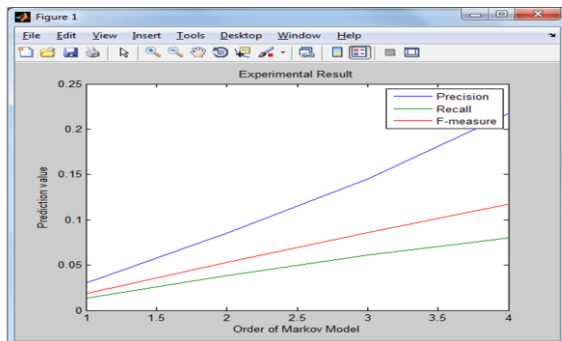
1 <sup>st</sup> order	A <sub>1</sub>	A <sub>2</sub>	A <sub>3</sub>	A <sub>4</sub>	A <sub>5</sub>	A <sub>6</sub>	A <sub>7</sub>	A <sub>8</sub>	A <sub>9</sub>	A <sub>10</sub>	A <sub>11</sub>	A <sub>12</sub>	A <sub>13</sub>	A <sub>14</sub>
S <sub>1</sub> ={A <sub>1</sub> }	0	0	0	2	0	1	0	0	0	1	0	0	0	0
S <sub>2</sub> ={A <sub>2</sub> }	0	0	0	1	1	1	1	0	0	0	0	0	1	0
S <sub>3</sub> ={A <sub>3</sub> }	0	0	0	1	0	0	1	1	0	0	0	0	0	0
S <sub>4</sub> ={A <sub>4</sub> }	1	1	1	0	0	1	0	1	0	0	0	1	1	0
S <sub>5</sub> ={A <sub>5</sub> }	0	0	0	0	0	0	0	0	1	0	0	0	0	1
S <sub>6</sub> ={A <sub>6</sub> }	1	1	0	2	0	0	0	0	0	1	1	0	0	0
S <sub>7</sub> ={A <sub>7</sub> }	0	2	0	0	0	0	0	0	0	1	0	2	0	0
S <sub>8</sub> ={A <sub>8</sub> }	0	0	0	0	0	1	0	0	0	2	0	1	0	0
S <sub>9</sub> ={A <sub>9</sub> }	0	0	0	0	1	0	1	0	0	0	0	0	0	0
S <sub>10</sub> ={A <sub>10</sub> }	0	0	0	1	0	1	0	1	0	0	0	0	1	1
S <sub>11</sub> ={A <sub>11</sub> }	0	0	0	0	0	0	0	0	0	0	0	0	0	0
S <sub>12</sub> ={A <sub>12</sub> }	0	0	0	0	0	1	0	0	1	0	0	0	0	0
S <sub>13</sub> ={A <sub>13</sub> }	0	0	0	0	1	0	0	1	0	0	0	0	0	1
S <sub>14</sub> ={A <sub>14</sub> }	0	0	1	0	0	0	0	0	0	1	0	0	0	0

Table 9: 2<sup>nd</sup> order transition probability matrix

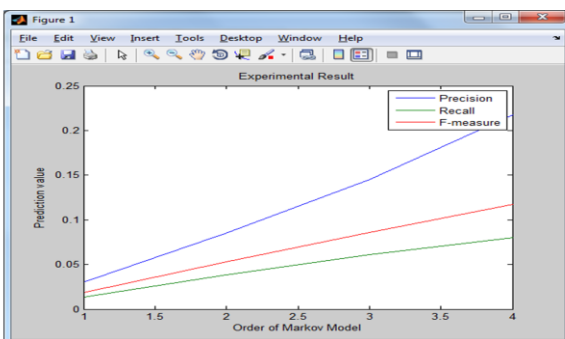
S. No.	2 <sup>nd</sup> order	A <sub>1</sub>	A <sub>2</sub>	A <sub>3</sub>	A <sub>4</sub>	A <sub>5</sub>	A <sub>6</sub>	A <sub>7</sub>	A <sub>8</sub>	A <sub>9</sub>	A <sub>10</sub>	A <sub>11</sub>	A <sub>12</sub>	A <sub>13</sub>	A <sub>14</sub>
1	{A <sub>1</sub> , A <sub>4</sub> }	0	0	0	0	0	0	0	1	0	0	0	1	0	0
2	{A <sub>1</sub> , A <sub>6</sub> }	0	0	0	1	0	0	0	0	0	0	0	0	0	0
3	{A <sub>1</sub> , A <sub>10</sub> }	0	0	0	0	0	0	0	1	0	0	0	0	0	0
4	{A <sub>2</sub> , A <sub>5</sub> }	0	0	0	0	0	0	0	0	0	0	0	0	0	1
5	{A <sub>2</sub> , A <sub>6</sub> }	0	0	0	0	0	0	0	0	0	1	0	0	0	0
6	{A <sub>2</sub> , A <sub>7</sub> }	0	0	0	0	0	0	0	0	0	1	0	0	0	0
7	{A <sub>2</sub> , A <sub>13</sub> }	0	0	0	0	1	0	0	0	0	0	0	0	0	0
8	{A <sub>3</sub> , A <sub>4</sub> }	0	0	0	0	0	1	0	0	0	0	0	0	0	0
9	{A <sub>3</sub> , A <sub>7</sub> }	0	0	0	0	0	0	0	0	0	0	0	1	0	0
10	{A <sub>3</sub> , A <sub>8</sub> }	0	0	0	0	0	1	0	0	0	0	0	0	0	0
11	{A <sub>4</sub> , A <sub>1</sub> }	0	0	0	0	0	0	0	0	0	1	0	0	0	0
12	{A <sub>4</sub> , A <sub>2</sub> }	0	0	0	0	0	0	1	0	0	0	0	0	0	0
13	{A <sub>4</sub> , A <sub>3</sub> }	0	0	0	0	0	0	0	1	0	0	0	0	0	0
14	{A <sub>4</sub> , A <sub>8</sub> }	0	0	0	0	0	0	0	0	0	1	0	0	0	0
15	{A <sub>4</sub> , A <sub>12</sub> }	0	0	0	0	0	1	0	0	0	0	0	0	0	0
16	{A <sub>4</sub> , A <sub>13</sub> }	0	0	0	0	0	0	0	1	0	0	0	0	0	0
17	{A <sub>5</sub> , A <sub>9</sub> }	0	0	0	0	0	0	1	0	0	0	0	0	0	0
18	{A <sub>5</sub> , A <sub>14</sub> }	0	0	0	0	0	0	0	0	0	1	0	0	0	0
19	{A <sub>6</sub> , A <sub>1</sub> }	0	0	0	1	0	0	0	0	0	0	0	0	0	0
20	{A <sub>6</sub> , A <sub>2</sub> }	0	0	0	1	0	0	0	0	0	0	0	0	0	0
21	{A <sub>6</sub> , A <sub>4</sub> }	1	0	0	0	0	0	0	0	0	0	0	0	1	0
22	{A <sub>6</sub> , A <sub>10</sub> }	0	0	0	1	0	0	0	0	0	0	0	0	0	0
23	{A <sub>7</sub> , A <sub>2</sub> }	0	0	0	0	1	0	0	0	0	0	0	0	1	0
24	{A <sub>7</sub> , A <sub>10</sub> }	0	0	0	0	0	0	0	0	0	0	0	0	0	1
25	{A <sub>7</sub> , A <sub>12</sub> }	0	0	0	0	0	0	0	0	1	0	0	0	0	0
26	{A <sub>8</sub> , A <sub>6</sub> }	0	0	0	0	0	0	0	0	0	0	1	0	0	0
27	{A <sub>8</sub> , A <sub>10</sub> }	0	0	0	0	0	1	0	0	0	0	0	0	0	0
28	{A <sub>9</sub> , A <sub>7</sub> }	0	0	0	0	0	0	0	0	0	0	0	1	0	0
29	{A <sub>10</sub> , A <sub>6</sub> }	0	1	0	0	0	0	0	0	0	0	0	0	0	0
30	{A <sub>10</sub> , A <sub>8</sub> }	0	0	0	0	0	0	0	0	0	0	0	1	0	0
31	{A <sub>10</sub> , A <sub>13</sub> }	0	0	0	0	0	0	0	0	0	0	0	0	0	1
32	{A <sub>12</sub> , A <sub>9</sub> }	0	0	0	0	0	1	0	0	0	0	0	0	0	0
33	{A <sub>13</sub> , A <sub>14</sub> }	0	0	1	0	0	0	0	0	0	0	0	0	0	0

## 8. EXPERIMENTAL RESULT

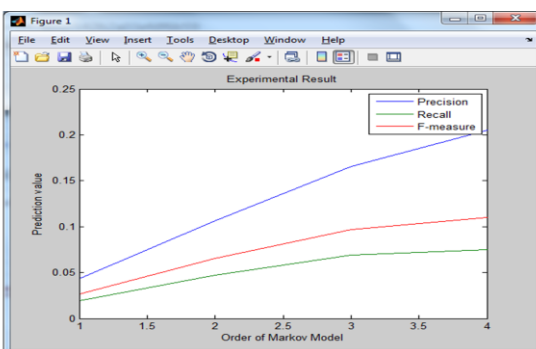
We have applied all the experiment in MATLAB tool. We have generated the dummy data set. The identification of session is also done. Predicted links are calculated and the result analysis of prediction is done by calculating precision, recall and f-measures. We can see the results here.



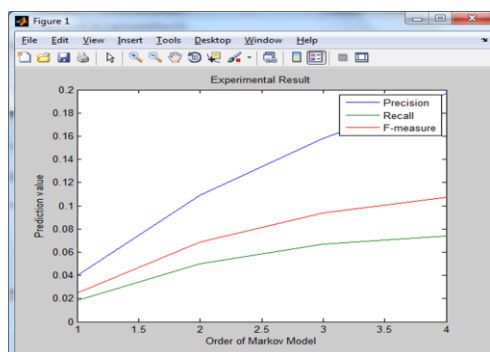
Data set size = 3000



Data set size = 6000



Data set size = 9000



Data set size = 15000

## 9. CONCLUSION

As we shown that we can use Markov model for recommending resources in social network. If we take higher order Markov model, then it takes lots of space to handle many states generated for predication but probability of accurate prediction is increased. As per our knowledge this is the first study in this field. We want to improve the accuracy of prediction. In future we want to apply Markov model for user behavior prediction and also for link prediction.

## 10. REFERENCES

- [1] Katz L, "A new status index derived from sociometric analysis," *Psychometrika*, 1953.
- [2] Lada A. Adamic and Eytan Adar, "Friends and neighbors on the web", *Social Networks*, 25(3):211-230, July 2003.
- [3] Zhou T, Lu L, Zhang Y, "Predicting missing links via local information", *The European*, 2009.
- [4] Clauset A, Newman M, Moore C, "Finding community structure in very large networks", *Physical Review E*, 2004.
- [5] Girvan M, Newman M, "Community structure in social and biological networks", *Proceedings of the National Academy of Sciences of the United States of America.*, 2002.
- [6] T. Joachims, D. Freitag, and T. Mitchell, "WebWatcher: A tour guide for the World Wide Web," in *Proc. IJCAI*, 1997, pp. 531-547.
- [7] O. Nasraoui and R. Krishnapuram, "One step evolutionary mining of context sensitive associations and Web navigation patterns," in *Proc. SIAM Int. Conf. Data Mining*, Arlington, VA, Apr. 2002, pp, 531-547.
- [8] M. Awad and L. Khan, "Web navigation prediction using multiple evidence combination and domain knowledge," *IEEE Trans. Syst. Man. Cybern. A. Syst., Humans*, vol. 37, no 6, pp, 1054-1062, Nov. 2007.
- [9] O. Nasraoui and C. Petenes, "Combining Web usage mining and fuzzy inference for Website personalization," in *Proc. WebKDD*, 2003, pp, 37-46.
- [10] Prem Melville and Vikas Sindhwani, *Recommender Systems*, Encyclopedia of Machine Learning, 2010.
- [11] Jump up^ R. J. Mooney and L. Roy (1999). "Content-based book recommendation using learning for text categorization". In *Workshop Recom. Sys.: Algo. and Evaluation*.
- [12] Adomavicius, G.; Tuzhilin, A. (June 2005). "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions". *IEEE Transactions on Knowledge and Data Engineering* 17 (6): 734–749. doi:10.1109/TKDE.2005.99.
- [13] J. Chen *et al.*, "Make new friends, but keep the old—Recommending people on social networking sites," in *Proc. ACM CHI*, Boston, MA, 2009.