International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 2 Issue 6, June-2013, pp: (71-75), Available online at: www.erpublications.com

# Discovery of frequent subgroup using data mining

Seema Mishra
Robotics and AI Lab, Indian Institute of Information Technology, Allahabad, India

**Abstract: Social network analysis is emerging technology that focuses on pattern of interaction/relation among people, organization. Frequent subgroup detection is subpart of social network that reveals the hidden knowledge and reflect the behavior of entire social network. These subgroups are collection of nodes that share common characteristics and densely connected with each other. In this paper, an unexampled approach is acknowledged to discover frequent subgroup inspired from a well known algorithm in the domain of association rule mining recognized as Continuous association rule mining algorithm.**

**Keywords: Subgroup detection, Social network analysis, Dynamic network analysis.**

## I. INTRODUCTION

In modern era, social network analysis has been in existence for quite some time and experiencing a surge in popularity to understand the behavior of the users in the form of nodes in the network. In order to model the social network, most popular data structure typically known as graphs are used where the nodes depict the individual or group of person, or event or organization etc and each link/edge represents connection/relationship between two individual [7 ,8]. Social network analysis attempts to understand the network and its components like nodes (social entities commonly known as actor or event) and connections (inter-connection, ties, and links). It has main focus of analyzing individuals and their relationships among them rather than individuals and their attributes as we deal in conventional data structure.
Social Network analysis has been in existence from past but now a day's extensively used to analysis the structure and connection between various actors existing within organization.

The ability to detect community structure in a network could have practical applications. Communities in a network might represent real social groupings oftentimes interacting, perhaps by interest or background; communities in a citation network might represent related papers on a single topic; communities in a metabolic network might represent cycles and other functional groupings; communities on the web might represent pages on related topics; hidden communities might represent potential suspicious activity.

Being able to identify these communities could help us understand and exploit these networks more effectively. Communities of practice are the collaboration groups that naturally grow and coalesce within any kind of networks. Any institution that provides opportunities for communication or interaction among its members is eventually threaded by communities who have similar goals and a shared understanding of their activities. These communities have been the subject of much research as a way to uncover the structure and interaction patterns within a network in order to understand the collective behavior of the network from the individuals that constitute the network. Recent Research on these networks has focused on using a social network perspective to analyze these networks.

A social network consists of both a set of actors, who may be arbitrary entities like persons or organizations, and one or more types of relations between them, such as information exchange or economic relationship. Subgroup detection aims at clustering nodes in a graph into subgroups that share common characteristics. But to some extent, sub graph discovery does the same job for finding interesting or common patterns in a graph. One of the most common interests of social network analysis is the substructures that may be present in the network. Subgroups are subsets of actors among whom there are relatively strong, direct, intense, frequent, or positive ties. From the ideas of subgroups within a network, we can understand social structure and embeddedness of individuals. Finding frequent groups in graph database can be modeled as (a) graph transaction setting and (b) single graph setting. Graph transaction setting takes as input relatively small graph of user interaction whereas single graph setting deals with large graph of user's interaction involved in communication [M. Kuramochi and G. Karypis, 2004].

The approach we espoused for discovering frequent subgroups is based on continuous association rule mining algorithm [15]. The main aim of adopting this approach is, because the groups of people is not static, it changes over a period of time as the member of group is being joining and leaving from group.

In order to model the interaction of users, we have an undirected graph where each vertex represents user and each edge represents relation between two users. Using such a graph representation, the problem of finding frequent patterns then becomes that of discovering subgroups which occur frequently enough over the entire set of graphs. The overall group of people is represented in figure1.

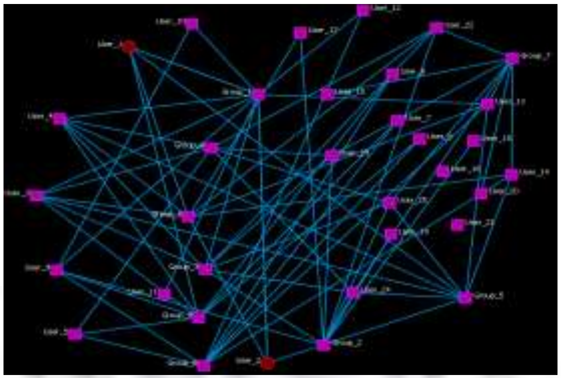

**Figure 1: Group of people**

For example, subgroups are represented in following figure2.

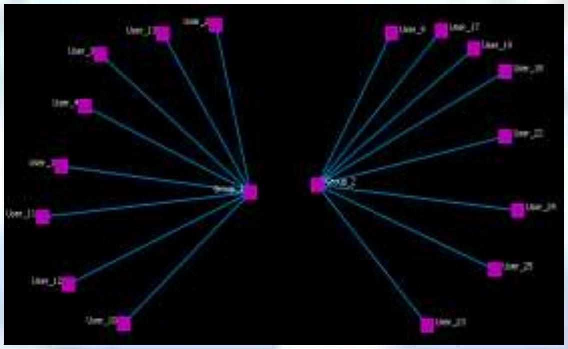

**Figure 2: Two subgroups**

## II. Overview of the Paper

Integration of paper is delimited as follows: In Section 3, we put a related work in the subgroup detection in social networking. In Section 4, CARMA algorithm will be discussed in the context of social network data. More appropriately will be said in the form of association /interaction among users. Section 5, incorporates insinuation of algorithm and discuss data sets and implementation results yielded by algorithm. This paper ends with conclusion in section 6.

## III. RELATED WORKS

Since network is structurally designed as consisting of nodes and edges indicate relationship. One special kind of network is known as social network and has been studying for long time [2, 3, 4, 5]. Modeling complex datasets in graph has been recognized powerful tool in various research domains like chemical domain [9, 10], computer vision [11], image and object retrieval [12], and machine learning [13]. In particular, Dehaspe et al. [10] applied Inductive Logic Programming (ILP) to obtain frequent patterns in the toxicology evaluation problem [14].

## IV. OVERVIEW OF ASSOCIATION RULE MINING IN THE CONTEXT OF CARMA

In order to search the group of people communicating frequently we followed a graph transaction setting model. There are two phases in algorithm as we have in CARMA. Phase 1 is meant to construct lattice of large group of people interacting /communicating each others. For each group sets $g \subseteq G$, three variables are asseverated:

**count(g):** the number of times group g occurred in communication while inserting in lattice.
**firstTrans(g):** pattern of communication number at which g is being inserted in lattice.
**maxMissed(g):** upper bound on the occurrence of g before g is being inserted in lattice.

## V. DATA SETS AND IMPLEMENTATION

This section exemplifies the approach. We considered a synthetic dataset incorporating Users Groups with following attributes of each user. Interaction of group is prepared in such an appropriate format that fits the experimental analysis

- age
- location
- sex
- education
- marital status
- interests

**Pattern of interaction:**

$g_1$ : ( user_1, user_2, user_3, user_4, user_10, user_11, user_12, user_13)

$g_2$ : ( user_22, user_23, user_24, user_25, user_17, user_18, user_19, user_9)

$g_3$ : ( user_22, user_23, user_24, user_25, user_17, user_18, user_19, user_9)

$g_4$ : ( user_1, user_2, user_3, user_4, user_22, user_23, user_24, user_25)

$g_5$ : ( user_1, user_2, user_3, user_4, user_13, user_14, user_24, user_25)

$g_6$ : (user_1, user_2, user_3, user_4, user_13, user_14, user_24, user_25)

$g_7$ : (user_1, user_2, user_3, user_4, user_22, user_23, user_24, user_25)

$g_8$ : ( user_3, user_4, user_5, user_22, user_23, user_24, user_25, user_10)

$g_9$ : ( user_13, user_14, user_15, user_22, user_23, user_24, user_25, user_20)

$g_{10}$ : (user_12, user_13, user_14, user_5, user_6, user_7, user_8, user_9)

$g_{11}$ : (user_12, user_13, user_14, user_15, user_24, user_25, user_1, user_2)

$g_{12}$ : (user_22, user_23, user_24, user_25, user_8, user_12, user_13, user_14)

$g_{13}$ : (user_1, user_2, user_3, user_11, user_12, user_13, user_24, user_25)

$g_{14}$ : (user_1, user_2, user_20, user_21, user_22, user_23, user_24, user_25)

$g_{15}$ :( user_8, user_9 user_10 user_11 user_12 user_13 user_23 user_24)

$g_{16}$ : (user_10, user_11, user_12, user_13, user_22, user_23, user_24, user_25)

$g_{17}$ : (user_10, user_11, user_12, user_13, user_22, user_23, user_24, user_25)

$g_{18}$ : (user_22, user_23, user_24, user_25, user_16, user_17, user_18, user_19)

$g_{19}$ : (user_22, user_23, user_24, user_25, user_7, user_18, user_21, user_11)

$g_{20}$ : (user_22, user_23, user_24, user_25, user_17, user_18, user_19, user_9)

$g_{21}$: (user_22, user_23, user_24, user_25, user_5, user_6, user_7, user_8)

$g_{22}$ : (user_1, user_2, user_3, user_4, user_22, user_23, user_24, user_25)

$g_{23}$ : (user_1, user_2, user_3, user_4, user_13, user_14, user_24, user_25)

$g_{24}$: (user_1, user_2, user_3, user_4, user_13, user_14, user_24, user_25)

$g_{25}$ : (user_1, user_2, user_3, user_4, user_22, user_23, user_24, user_25)

the support sequence is supplied as
$\sigma$ =[0 1 2 2 3 3 3 3 3 4 4 4 4 5 5 5 5 5 5 5 6 6 6]
The group of people communicating frequently using algorithm is resulting as
(user_1, user_2, user_3, user_4, user_10, user_11, user_12, user_13)

Each user is linked with some communities of their own interest on the basis of that we made groups of users having some specific values for each attribute. Interaction of group is defines as pattern of communication in different groups $g_{1,}g_{2,}g_{3,\ldots}g_n$ and support lattice of users group is G.

For the sake of convenience and ease of explanation for how algorithm works, we deducted a small part of user pattern of interaction as mentioned P = { $g_1, g_2, g_3$ } where people involved in these groups as:

$g_1$ = {u1, u2, u4}

$g_2$ = {u1, u2, u3}

$g_3$ = {u1, u2}

All the way of implementation (shown in figure 3), G is initialized to $\{\phi\}$ and corresponding integers as (0, 0, 0) and for easy calculation support sequence is $\sigma = (0.3, 0.9, 0.7)$.

Since maxSupport ( $\phi$ ) =1≥ $\sigma_1(0.3)$, add all singleton users and set associated integers with updated values as (0, 1, 1). User sets are not pruned in first scanning. Now scanning of $g_2$, since {u1, u2} $\subseteq g_2$ and {u1}, {u2} already in G with maxSupport=1 which is greater than $\sigma_2$. So insert {u1, u2} is inserted to G with updated associated integers.

**(maxMissed, firstTrans, count)**
**[minSupport, maxSupport]**

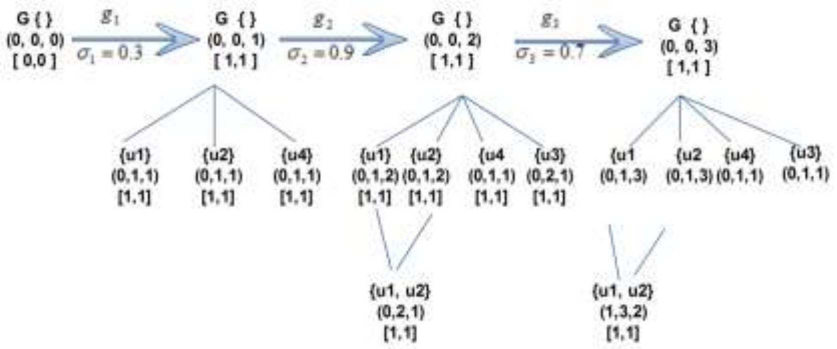

**Figure 3: CARMA implementation of lattice of group**

## VI. CONCLUSION

In this paper, we presented frequent group detection algorithm based on CARMA. It is capable of handling dynamism of network as members are joining or leaving the group. We carried out the implementation on synthetic data of user interaction.

## VII. AKNOWLEDGEMENT

We would like to thank Mr. Lokesh Kumar and Mr. Naveen B. Tech student of RGIIT Campus, Amethi for his endeavor in implementing algorithm.

Organizational Risk Analyzer (ORA) tool available through the Center for Computational Analysis of Social and Organizational Systems (CASOS), http://www.casos.cmu.edu., is used to draw graphs.

## REFERENCES

[1] M. Kuramochi and G. Karypis. Finding frequent patterns in a large sparse graph. SDM, 2004.

[2] Newman, M. E. J. Detecting community structure in networks. European Physical Journal B 38: 321-330. 2004.

[3] Newman, M. E. J Fast algorithm for detecting community structure in networks. Physical Review E 69: 066133, 2004.

[4] Luo, J. Social network analysis. Social Science Academic Press. (In Chinese), 2004.

[5] Wasserman, S. and K. Faust. Social network analysis: Methods and applications. New York, Cambridge University Press, 1994.

[6] Girvan, M. and M. E. J. Newman. Community structure in social and biological networks. Proceedings of the National Academy of Sciences of the United States of America 99: 7821–7826, 2002.

[7] Hanneman, Robert A. and Mark Riddle. Introduction to social network methods. Riverside, CA: University of California (published in digital form http://faculty.ucr.edu/~hanneman/, 2005.

[8] H. Goto, Y. Hasegawa, and M. Tanaka, “Efficient Scheduling Focusing on the Duality of MPL Representatives,” Proc. IEEE Symp. Computational Intelligence in Scheduling (SCIS 07), IEEE Press, Dec. 2007, pp. 57-64, doi:10.1109/SCIS.2007.357670.