SOCIAL NETWORK IDENTIFICATION AND ANALYSIS

USING CALL DETAIL RECORDS

Thesis Supervisor: Dr Shoab Ahmed Khan

Student Name: Muhammad Usman

2005-NUST-MSc PhD-ComE-07

College of Electrical and Mechanical Engineering

National University of Science and Technology

Pakistan

2007

DEDICATION

In the name of Allah, the most Gracious, the Most Merciful

To my parents and teachers, without whose unflinching support and unstinting cooperation, a work like this would not have been possible

ACKNOWLEDGEMENT

I would like to take this opportunity to express great indebtedness and gratefulness to Almighty Allah who gave me the vigor and determination to complete this thesis. I greatly recognize the continuous supervision and motivation provided by my advisor, Dr. Shoab Ahmed Khan. I would also like to thank my committee members, Brig Dr Muhammad Younus Javed, Dr Shleza Sohail and Ms Aasia Khanum for their ever-extended moral and technical support.

Abstract

This thesis proposes comprehensive solution analysis of call detail records for discovering and analyzing social networks and prominence of an actor in the network. It is based on all the existing techniques and methodologies for telecom billing solutions. On the basis of existing systems database has been developed in SQL server 2005. The front end has been developed in Visual Studios 2005 in Visual C# environment. All the basic functionalities have been provided for billing systems. The basic functionalities include searching through CDR and generating required reports from the database. Also revenue generating customers have been identified. Finally methodologies have been proposed for identification and analysis of social networks from CDR.

These methodologies include grouping the related mobile users using CDR's by selecting a central node. Also networks analysis has been made on the basis of degree centrality and prestige to find the prominence of a user in the group or whole network or in cluster. Algorithms have been proposed for clustering and for network analysis.

Finally simulation for the working of the grouping is developed. This simulation is quite user friendly. It shows the link information between actors in a group or cluster in quite a comprehensive way. It explains most of the functionality with sociogram generation.

Table Of Contents

List of Figures

Chapter 1

Introduction

A social network represents a group of people (or organizations or other social entities) connected through socially-meaningful relationships, such as friend, co-workers, or information exchange [13]. Garton and colleagues [12] synthesize literatures in past years and list four units of analysis: relations, ties, multiplicity, and composition. There are two main properties of networks and actors: connection and distance. In the aspect of connection, more connections often mean that individuals are exposed to more diverse information. Individuals highly connected may be more influential, and may be more influenced by others. Disease, rumors and useful information spread more quickly where there are high rates of connection. Size, density, degree, reach ability, centrality and prestige are commonly used for representing connectivity. The idea of the distance between actors represents how close they are to one another. It can help us to understand diffusion, homogeneity, solidarity, and other differences in macro properties of social groups.

Where distances are great, it may take a long time for information to diffuse across a population. It may also be that some actors are quite unaware of and influenced by others, even if they are technically reachable, the costs may be too high to conduct exchanges. Walk, trail and path are basic concepts to develop more powerful ways of describing various aspects of the distances among actors in a network.

A social network has a set of relations of ties, which can be viewed in two different ways. One view, called ego-centered network, focuses on an individual, and puts him or her at the network center. The other view focuses on the whole network based on some specific criterion of population boundaries such as a formal organization, department, club or kinship group. Subgroups are subsets of actors among whom there are relatively strong, direct, intense, frequent, or positive ties. From the idea of subgroups within a network, we can understand social structure and the embedded ness of individuals. In this research, we only use connected components to identify and analyze social networks by using call detail records.

The telecommunication industry has grown tremendously within the past few years. The number of subscribers is increasing day by day so it is becoming more and more difficult to monitor the activity of users in the mobile network. All the subscribers are not playing important role in the network and mobile companies might need to identify the valued customers for offering new services or promotions of existing services. Also criminal network analysis currently is primarily a manual process, usually consuming much time and human effort at each stage of the knowledge discovery process. So some tool is needed to identify the set of connected users (subnets) in the mobile network (clustering) which can be used for criminal analysis.

1.1. Problem Definition

The telecommunications industry generates and stores a tremendous amount of data. These data include call detail data, which describes the calls that traverse the telecommunication networks, network data, which describes the state of the hardware and software components in the network, and customer data, which describes the telecommunication customers. The amount of data is so great that manual analysis of the data is difficult, if not impossible. Call data can be used to construct customer networks and analyze how customers are related to each other [5]. The Call detail records not only contain the originating and terminating numbers but also the date, duration and cost of the call. The only problem is call detail records are not maintained separately for every user instead it contains the records of all the users. In this research methods have been proposed to analyze communication data and discover the network structures by carrying out social network analysis on call detail records.

1.2. Challenges and Constraints

The problem description presented provides a general outline, ignoring the technical details of the task and focusing on the goals of the thesis. Examining the problem in more detail, a number of challenging issues and problem features come to light, which have restricted the type of solution being sought.

1.2.1. Fuzzy Boundaries

Fuzzy boundaries occur because of Continuousness, when the data is not discrete, Aggregation, where discrete boundaries represent the average location of continuous or discrete variables binned together for descriptive convenience, usually categorized by comparison with a prototype, averaging, where discrete boundaries average a single time or scale varying geographical boundary, Imprecision, where boundaries are not known because they cannot be measured accurately enough, Ambiguity, where boundaries are tied to decision factors.

While searching the network structure in data, the main difficulty is in deciding which actor to include and which not to include. This problem is catered by making decision at each node or actor of the network.

1.2.2. Dynamic

Social network research has made extensive use of visualization. Actors are usually represented as points, and relations among actors are represented by lines, with relational direction indicated by arrows. Early network graphs were drawn by hand, and the layout was determined by the artistic and analytic eye of the author. Such early graphs were usually simple, having few relations per person or a clear hierarchical structure. The state of the art has progressed remarkably, and a growing body of research has developed around various definitions for optimal network layout. Most network images do a poor job of representing change in networks, and researchers make do by presenting successive snapshots of the network over time. To effectively display the relational structure of a social network, at least two dimensions are needed to represent proximity, and that leaves no effective space to represent time. However, recent advances make use of space to represent social distance and movement to represent change over time.

The hidden networks are not static, they are always changing. This requires constant network monitoring.

1.3. Aim

Analysis of call detail records for discovering and analyzing Social networks and Prominence of an actor in the network.

1.4. Objectives

On the basis of the problem statement following objectives has been identified.

a. Identifying Valued Customers on the basis of revenue generated.
b. Identifying Immediate Group by exploring of directly connected users.
c. Identifying Clusters by exploring all the directly or indirectly connected users.
d. Graphical Representation of group or cluster.
e. Degree Centrality and Prestige Measures to find the prominence of an actor in a group, cluster or whole network.

1.5. Organization of Thesis

Chapter 1: Introduction. It briefly explains the need and importance of research work. It also explains the problem definition, aim and objectives of the thesis.

Chapter 2: Literature Review: it gives details about the literature consulted during the thesis, mainly comprising the social networks related study carried out. It also briefly explains the existing systems and the previous work carried out in this area.

Chapter 3: System Design and Implementation. This chapter explains the system design and requirements and implementation of the proposed system. It explains the developed system modules along with data flow diagrams, algorithms and entity relationship diagram.

Chapter 4: Results and Analysis. This chapter explains the proposed system with the help of examples. It also gives the conclusions, areas of application and future extensions for the proposed system.

Chapter 2

Literature Review

2.1. Social Network

A social network can be defined as any bounded set of connected people.

2.1.1. Characteristics of Social Networks

A network can be defined as any bounded set of connected units.

2.1.1.1. Bounded

Networks are bounded in the sense that some criteria can be set to determine membership of a unit in the network. In simple networks, the boundary criteria is relatively simple to define, for example, a family network consists of members of a family or players belonging to a specific football team but as the complexity of the networks increases it becomes a critical step to set a network's boundaries.

2.1.1.2. Connected

Connectedness constitutes the basic definition of a network. To be part of a network, each member must have links to at least one other member of the network. These links may be direct or indirect. While some members may be peripheral in the network or almost completely isolated, each one must somehow be connected to other members if it is to be considered part of the network. This depends also on what criteria are selected for defining the network boundaries.

2.1.1.3. Network Units

The third aspect of a network is defining a unit. Network analysis can be easily applied to a wide range of units. In social network analysis units can be individuals, organizations, or even whole nations in case of global networks or units can also be phone numbers, email accounts, or a discussion thread.

2.1.2. Representation of Social Networks

Different approaches, with varying complexity and focus, have been proposed to represent interactions among individuals in a way so as to increase understanding of the network they form. The two basic concepts are graph models and matrix models.

2.1.2.1. Graph Representation

Network analysis has depended on the way the network is visualized since the time network analysis has been conceived. Klovdahl [3] has considered points and lines as a natural way of representing a network. Moreno [4], represented social networks as sociograms, which display the relations among network members in a two-dimensional space. Members of the network are represented as points or nodes, with lines drawn between pairs of nodes to show a relationship between them. An arrow is sometimes used to show the direction of flow in a relationship.

One of the main disadvantages of displaying networks, as graphs, is that a slight change in the visual image of the graph can give an entirely new picture of the structure of the underlying network. Since, a network maybe represented by a number of graph drawings, which are logically correct and represent the same underlying network, it becomes difficult to limit oneself to one underlying assumption of a drawing. Also, as the number of members and the number of connections between members increase, interpretation of the diagram becomes increasingly difficult. For large, densely connected networks, visual displays can be so complex that they confuse the understanding of the network's structure. Related to the problem of interpretation is the difficulty of producing graphs of large networks. Traditionally, network sociograms were drawn by hand. Such approaches are more like an art form than an advanced analysis technique. There are few generally accepted procedures for developing such graphic presentations. Even with the advances in computer graphics, large complex network graphs have been slow to evolve. In some cases, plotting programs developed for other purposes have been used to create graphic depictions of networks.

2.1.2.2. Matrix Representation

The use of matrices has become the dominant approach to network analysis in recent years. It produces an algebraic representation of network relations. While a matrix does not stimulate the kind of intuitive understanding that a simple\ graph model does, it has nonetheless become prominent because it expresses all the information pictured in a graph model. Moreover, it also facilitates more extensive quantitative analysis.

A matrix represents a network in the form of an array of units arranged in rows and columns. In a typical network matrix, the rows represent network members while the columns represent the attributes of the network member units and the cell values indicate the measurement of a particular attribute for a particular member unit. In the case of adjacency matrix, the columns are the set of same member units and each cell in the matrix contains a number that represents the relationship between two members of the network. Usually a ‘1’ represents a relationship between two members and a ‘0’ represents the absence of a relationship.

Matrices can represent both directional and non]directional relations among the network members. With directional relations, the members arrayed in the matrix rows are typically treated as initiators or senders of the content in a relationship, and the members arrayed across the columns are viewed as recipients of the content in a network.

If directional relationships in a network are not completely reciprocal, the matrix is asymmetrical. That is, the number in row *i* and column *j* is not identical to the number in the row *j* and column *i*. The matrix is symmetrical when directional relationships are reciprocal, or when the data represents non] directional\ relationships. Then, the number for every link from *i* to *j* is also represented in the link from *j* to *i*.

2.1.3. Social Network Analysis

Social network analysis (SNA) is the mapping and measuring of relationships and flows between people, groups, organizations, computers, mobiles or other information processing entities.

Network analysis is becoming increasingly popular as a general methodology for understanding complex patterns of interaction. While the network concept has deep roots in anthropology and sociology [1], standard techniques for studying the structure social networks have been relatively recent developments. Several factors contribute to the growing popularity of social network analysis. Firstly, the world is becoming more interdependent as reflected in overlapping corporate boards, international markets, specialized service economies, and the involvement of multiple levels of government in many aspects of daily life.

Another factor is the applicability of social network analysis across different units and levels of analysis. As Burt [2] points out, social network analysis is a potentially powerful methodology for connecting micro and macro levels of social theory. A third factor has been advances in computer technology which makes it possible to design network studies and conduct complex network analyses which were impossible just a decade ago.

2.1.3.1. Network Metrices

Network Metrices are used to check the prominence of an actor in a group. There are a lot of metrics to analyze a network but I have considered only the degree Centrality and degree Prestige.

2.1.3.1.1. Degree Centrality

Degree centrality is the measure of the activity in the network. The degree centrality is defined based on only the out degree (outgoing calls), the number of out-links or edges denoted by *do(i)*. The degree of directed graph is given by following equation.

$$C'_D(i) = \frac{d_o(i)}{n-1}$$

2.1.3.1.2. Degree Prestige

Prestige is another measure of link and it is a more refined measure of prominence of an actor than centrality. A prestigious actor is one who is object of extensive ties as a recipient. To compute the prestige, only in-links are used. Based on the definition of the prestige, it is clear that an actor is prestigious if it receives many in-links. Thus the simplest measure of prestige of an actor i, denoted by Pd(i), is its in-degree given by following equation.

$$P_D(i) = \frac{d_I(i)}{n-1}$$

2.2. Call Detail Record VS Social Network Analysis

CDR provides communication network information operating on a computer network that supports social networks. They combine location flexibility, rapid transmission to multiple others across time. As an indicator of collaboration and knowledge exchange, CDR provides a rich source for extracting informal social network information from mobile communication across networks.

CDR are rich in social information, time signatures of sent, received as well as other types of communication behavior of its user such as how often the user calls at certain time of day. CDR offers an enormous amount of information on its user’s behavior.

2.3. Related Work

H.L.Larsen and N.Memon in [6] proposed algorithms for constructing hierarchy of the covert networks, so that investigators can view the structure of the ad hoc networks/ atypical organizations, in order to destabilize the adversaries. Moreover they also demonstrate techniques for filtering graphs (networks) / detecting particular cells in adversary networks using a fictitious dataset. In this research they simply find outs the terrorist networks by filtering the link information of specific actor in the network. The authors have used the centrality measure to analyze a set of actor in a tree. They have given algorithms to make tree from undirected or directed graphs but they haven’t proposed any system to discover or analyze clusters. This paper written specifically in relation with SNA but doesn’t contain any information to process CDR.

Christine, Andreas and Martin [7] explained that the telecommunication industry accumulate huge amounts of data not only about the usage behavior of individual customers, but also about how customers interact. They suggested that In addition to traditional data mining and statistical techniques, methods from the field of Social Network Analysis (SNA) are essential to leverage this special set of data. In this paper, they have compared different centrality measures based on a variety of different network topologies and model assumptions. The interesting result in their research was that independent form the parameters in all experiments out-degree-centrality performed the best.

Lian, Michael and Patrick in [9] used delayed CDR data as the primary data source to predict customer behavior. They extracted calling links, i.e., who called whom, from the CDR data, and proposed several distance measures based on calling links. They demonstrated that, by using information derived from the calling links alone as inputs to a neural network model, an acceptable accuracy for predicting churn (customer switching from one service provider to another) can be achieved. Calling links can also be used to identify calling communities, which may be used for targeted marketing campaigns and help predict acceptance of marketing offers.

Teng and Chou in [10] proposed that the communities of acquainted mobile users can be effectively discovered from collected CDRs and understanding the communities and corresponding calling behaviors are of great importance to telecommunication companies. Their study showed that their proposed approach is practically feasible.

All these research focuses on social network analysis to discover and analyze the group of some specific actor which actually contains only the in-links and out-links of that actor which I have called in my research as immediate group. In [7] the authors deduced that degree centrality is the best for analysis but they haven’t discussed or made any calculations for degree prestige which basically focuses on the in-links. Also there are no details for discovering the whole set of directly or indirectly connected actors.

In my research I have considered discovering and analyzing the immediate group as well as clusters. I have also made network analysis for whole network on the basis of degree centrality and degree prestige to find out the valued customers and the prominence of users in particular group or cluster. Finally I have created sociograms for groups and clusters.

Chapter 3

System Design and Implementation

3.1. System Overview

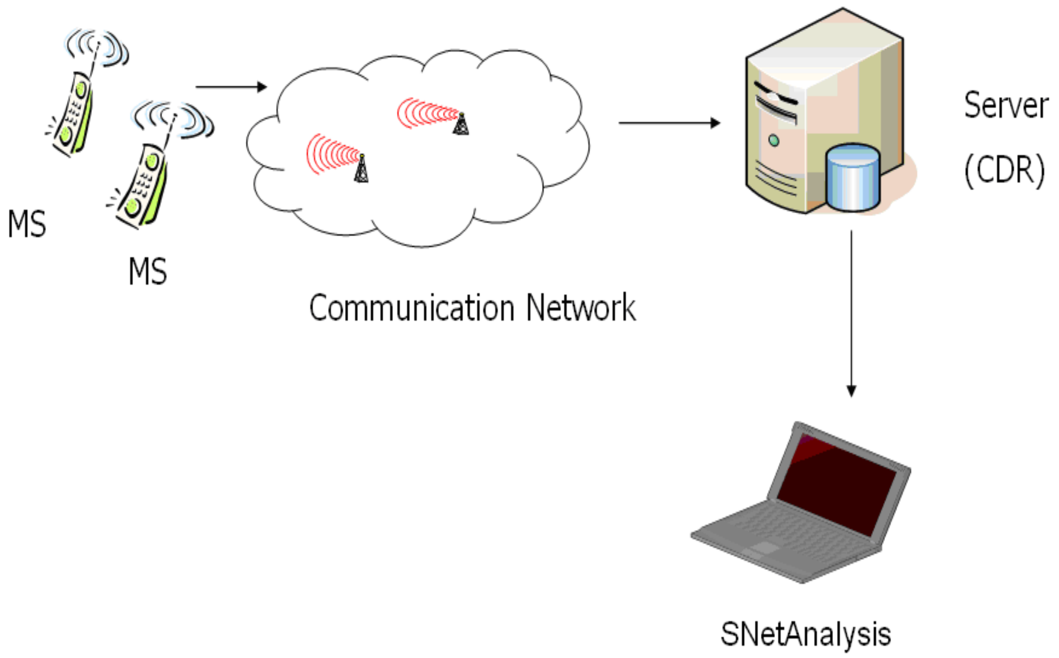


Figure 3.1 System Overview

Every time a call is placed on a telecommunications network, descriptive information about the call is saved as a call detail record. The number of call detail records that are generated and stored is huge. Call detail records include sufficient information to describe the important characteristics of each call. At a minimum, each call detail record will include the originating and terminating phone numbers, the date and time of the call and the duration of the call. Call detail records are generated in real-time and therefore will be available almost immediately for data mining. Figure 3.1 briefly describes the working of system. The generated CDR is ultimately stored in database from where the SNETAnalysis software developed in this thesis can access records and make the analysis.

3.2. Database Development

The database for call detail record has been developed in Microsoft SQL server 2005 Mobile Edition.

3.2.1. Requirements Specification

There are several requirements on the basis of which database has been developed. These requirements are

- The records for outgoing calls must be maintained for the customers belonging to the mobile network and there should be no outgoing call record if the caller is not a part of that network.
- Bills of each customer should be maintained if he has one or more call detail records.
- All the records must contain field like date of insertion or updating so queries should be made on monthly or yearly basis.
- Methods should be developed to provide functionalities such as searching through records for specific customer, finding out the most important customers on the basis of some factor like revenue, network analysis should be done on the basis of database and graphs should be made to show the interlinked customers.

3.2.1.1. Entities

On the basis of above requirements following entities can be defined in the network

- Customers
- Call Detail Records
- Bills
- Summarized call detail records

3.2.1.2. ER Diagram

ER Diagram is basically entity relationship diagram which explains the links between tables of the relational database. The following ER Diagram can be made on the basis of above mentioned assumptions.

The detailed ER Diagram made from SQL server 2005 is as follows.

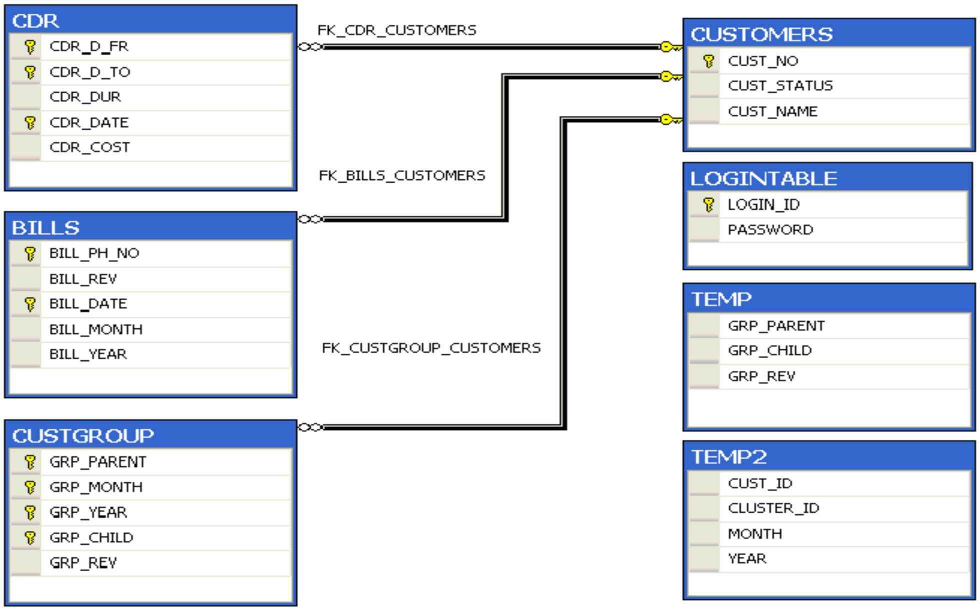


Fig 3.2 Entity Relationship Diagram

3.2.2. Database Development Details

The database required in this thesis needs Call Detail Records which are composed of incoming call logs and outgoing call logs with date and time stamps. On the basis of these CDRs we can extract the required information.

The CDR database contain a collection of tables which are as follow

3.2.2.1. Customer Table

3.2.2.2. Call Detail Record Table

3.2.2.3. Bills Table

3.2.2.4. Caller Groups Table

3.2.2.5. Temporary Table

3.2.2.1. Customer Table (CUSTOMERS)

This table contains the following columns

- Customer number (CUST_NO)

  Its 11 digits long which stores the mobile / customer number assigned by the operator.

- Customer Name (CUST_NAME)

  Its 50 character long which stores full name of the customer.

- Customer Status (CUST_STATUS)

  It's a single bit which indicates whether the number assigned to the user is activated or deactivated.

3.2.2.1.1. Relationships

This table is one to many relationship with CDR, Bills and Caller Group Table.

3.2.2.1.2. Table Creation Query

```
//Creates Table named CUSTOMES with coulumn names CUST_NO,
//CUST_STATUS, CUST_NAME.
CREATE TABLE [dbo].[CUSTOMERS](
        [CUST_NO] [numeric](19, 0) NOT NULL,
        [CUST_STATUS] [char](10) NOT NULL
        [CUST_NAME] [char](25) NOT NULL,
//Defining CUST_NO field as Primary Key to uniquely identify the entries
 CONSTRAINT [PK_PhoneNumbers] PRIMARY KEY CLUSTERED
(
        [CUST_NO]
)
```

3.2.2.2. Call Detail Record Table (CDR)

This table contains the following columns

- Call Initiation Number (CDR_D_FR) FK

  This field stores the number of customer from which call has been initiated.

- Call Accept Number (CDR_D_TO)

  This field stores the number of customer who receives the call which is initiated from CDR_D_FR.

- Call Duration (CDR_DUR)

  This field stores the total duration of call in seconds.

- Call Cost (CDR_COST)

This field stores the cost of the call and is calculated on the basis of formula specified in column computed specification.

- Call Date and Time (CDR_DATE)

This field stores the date and time of call.

3.2.2.2.1. Relationship

This table has 1 : M relationship from customer table to CDR table where CUST_NO from customer table acts as Foreign Key for CDR_D_FR for CDR Table.

3.2.2.2.2. Table Creation Query

```
//Creates Table named CDR with coulumn names CDR_D_FR, CDR_D_TO,
//CDR_DUR, CDR_DATE, CDR_COST
CREATE TABLE [dbo].[CDR]
(       [CDR_D_FR] [numeric](19, 0) NOT NULL,
        [CDR_D_TO] [numeric](19, 0) NOT NULL,
        [CDR_DUR] [numeric](10, 0) NOT NULL,
        [CDR_DATE] [datetime] NOT NULL,
        [CDR_COST]  AS ([CDR_DUR]*(0.05))
//Defining Primary Key as Composite Key consisting of CDR_D_FR
//(CUST_NO as Foreign Key), CDR_D_TO, CDR_DATE to uniquely identify
//the entries
        CONSTRAINT [PK_CDR] PRIMARY KEY CLUSTERED
        (       [CDR_D_FR] ,
                [CDR_D_TO] ,
                [CDR_DATE]
        )
) ON [PRIMARY]
```

3.2.2.2.3. Triggers

This table also contains trigger for updating bills and caller group table for every inserted entry in CDR table which is as follow.

```
ALTER TRIGGER [dbo].[CDRTOBILL] ON [dbo].[CDR]  FOR INSERT AS
DECLARE @nDFR numeric(13) // Variable to Store CDR_D_FR
DECLARE @nDTO numeric(13) // Variable to Store CDR_D_TO
DECLARE @nCOST FLOAT(9) // Variable to Store CDR_COST
DECLARE @dDAT CHAR(50) // Variable to Store CDR_DATE
// Storing CDR_D_TO, CDR_D_FR,CDR_COST,CDR_DATE from inserted
//entry in CDR table to above defined variables
SELECT @nDFR = [CDR_D_FR] FROM INSERTED
SELECT @nDTO = [CDR_D_TO] FROM INSERTED
SELECT @nCOST = [CDR_COST] FROM INSERTED
SELECT @dDAT = [CDR_DATE] FROM INSERTED
//If CDR_D_TO, is already present in BILLS table compare the
//BILLS_MONTH and BILLS_YEAR fields with CDR_DATE field from dDAT
//variable and if the statement is true update that entry by updating the
//BILL_REV and BILL_DATE fields in the BILLS table using variable values
IF (SELECT BILL_MONTH FROM BILLS WHERE BILL_PH_NO=@nDFR
AND BILL_MONTH = MONTH(@dDAT) AND BILL_YEAR=YEAR(@dDAT) )
= MONTH(@dDAT)
```

```
BEGIN
UPDATE  [BILLS]
SET BILL_REV = @nCOST+BILL_REV,
BILL_DATE=@dDAT
FROM BILLS WHERE BILL_PH_NO= @nDFR AND
BILL_MONTH=MONTH(@dDAT) AND BILL_YEAR = YEAR (@dDAT)
END
//If the statement is false insert new entry by using values in the variables in
//the BILLS table
ELSE
BEGIN
INSERT INTO BILLS (BILL_PH_NO,BILL_REV,BILL_DATE) VALUES
(@nDFR,@nCOST,@dDAT)
END
//If CDR_D_TO, is already present in CUSTGROUP table compare the
//GRP_PARENT,GRP_CHILD,GRP_MONTH,GRP_YEAR fields with values in
//variables and if the statement is true update that entry by updating the
//GRP_REV field in the CUSTGROUP table using variable values
IF (SELECT GRP_MONTH FROM CUSTGROUP WHERE
GRP_PARENT=@nDFR AND GRP_CHILD = @nDTO AND
GRP_MONTH = MONTH(@dDAT) AND GRP_YEAR=YEAR(@dDAT) ) =
MONTH(@dDAT)
BEGIN
UPDATE  [CUSTGROUP]
SET GRP_REV = @nCOST+GRP_REV
```

```
FROM CUSTGROUP
WHERE GRP_PARENT= @nDFR AND GRP_CHILD = @nDTO AND
GRP_MONTH=MONTH(@dDAT) AND GRP_YEAR = YEAR (@dDAT)
END
//If the statement is false insert new entry by using values in the variables in
//the CUSTGROUP table
ELSE
INSERT INTO CUSTGROUP
(GRP_PARENT,GRP_MONTH,GRP_YEAR,GRP_CHILD,GRP_REV)
VALUES (@nDFR,MONTH(@dDAT),YEAR(@dDAT),@nDTO,@nCOST)
```

3.2.2.3. Bills Table (BILLS)

This table contains the following columns

- Billing number (BILL_PH_NO)

  This field stores the customer number.

- Revenue generated (BILL_REV)

  This field stores the total bill of customer specified in BILL_PH_NO for particular month.

- Bill Date (BILL_DATE)

  This field stores the date and time at which bill was updated.

- Bill Month (BILL_MONTH)

  This field stores the billing month as integer from 1 to 12.

- Bill Year (BILL_YEAR)

  This field stores the billing year as integer.

This table is updated automatically for every inserted entry in CDR table. This procedure is defined in the triggers for CDR table.

3.2.2.3.1. Table Creation Query

```
//Creates Table named BILLS with coulumn names BILL_PH_NO,
//BILL_REV, BILL_DATE, BILL_MONTH, BILL_YEAR
CREATE TABLE [dbo].[BILLS]
(       [BILL_PH_NO] [numeric](19, 0) NOT NULL
        [BILL_REV] [numeric](19, 4) NULL
        [BILL_DATE] [datetime] NULL,
        [BILL_MONTH]  AS (datepart(month,[BILL_DATE])),
        [BILL_YEAR]  AS (datepart(year,[BILL_DATE]))
//Defining Primary Key as Composite Key consisting of BILL_PH_NO
//(CUST_NO as Foreign Key),BILL_DATE to uniquely identify the entries
        CONSTRAINT [PK_BILLS] PRIMARY KEY CLUSTERED
        (       [BILL_PH_NO] ,
                [BILL_DATE]
        )
) ON [PRIMARY]
```

3.2.2.4. Caller Group Table (CUSTGROUP)

This table basically maintains the adjacent list for the whole network which is further used to make group among users.

It contains the following fields

- Group Parent (GRP_PARENT)

It stores the number of parent or root node.

- Group child (GRP_CHILD)

It stores the child or sub nodes number for particular parent.

- Group revenue (GRP_REV)

If stores the total revenue generated by calling from parent to child for particular month.

- Month (GRP_MONTH)

It stores the month of current updated entry.

- Year (GRP_YEAR)

It stores the year of current updated entry.

This table is also updated automatically for every inserted entry in CDR table and the procedure is defined in the triggers for CDR table.

3.2.2.4.1. Table Creation Query

```
//Creates Table named CUSTGROUP with coulumn names GRP_PARENT,
//GRP_CHILD, GRP_MONTH, GRP_YEAR, BILL_REV
CREATE TABLE [dbo].[CUSTGROUP]
(
        [GRP_PARENT] [numeric](19, 0) NOT NULL,
        [GRP_MONTH] [int] NOT NULL,
        [GRP_YEAR] [int] NOT NULL,
        [GRP_CHILD] [numeric](19, 0) NOT NULL,
        [GRP_REV] [numeric](19, 4) NOT NULL
)
```

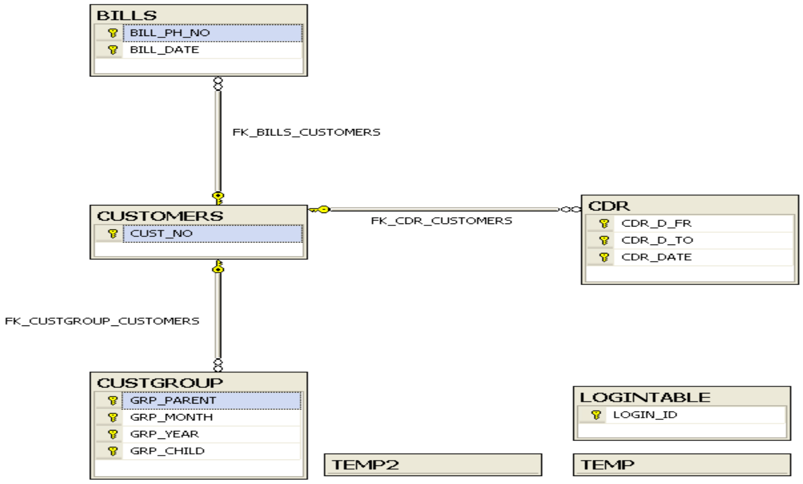
```
//Defining Primary Key as Composite Key consisting of GRP_PARENT
//(CUST_NO as Foreign Key), GRP_CHILD, GRP_MONTH, GRP_YEAR to
//uniquely  identify the entries
        CONSTRAINT [PK_CUSTGROUP] PRIMARY KEY CLUSTERED
        (       [GRP_PARENT] ,
                [GRP_MONTH] ,
                [GRP_YEAR] ,
                [GRP_CHILD]
        )
) ON [PRIMARY]
```

3.2.2.5. Temporary Table

These Tables named temp1 and temp2 contains no records or information in fact they are used by the GUI application to compute results for analysis and sociogram generation for users.

3.2.3. Integrity Constraints

Enforcing data integrity guarantees the quality of the data in the database. For example, if an employee is entered with an employee_id value of 123, the database should not permit another employee to have an ID with the same value. If you have an employee_rating column intended to have values ranging from 1 to 5, the database should not accept a value of 6. If the table has a dept_id column that stores the department number for the employee, the database should permit only values that are valid for the department numbers in the company.



3.3. Diagram Showing Integrity Constraints

3.3. Software development

The software has been developed in Microsoft Visual Studios 2005 in C #. Microsoft Visual C#, pronounced C sharp, is a programming language designed for building a wide range of applications that run on the .NET Framework. C# is simple, powerful, type-safe, and object-oriented. With its many innovations, C# enables rapid application development while retaining the expressiveness and elegance of C-style languages.

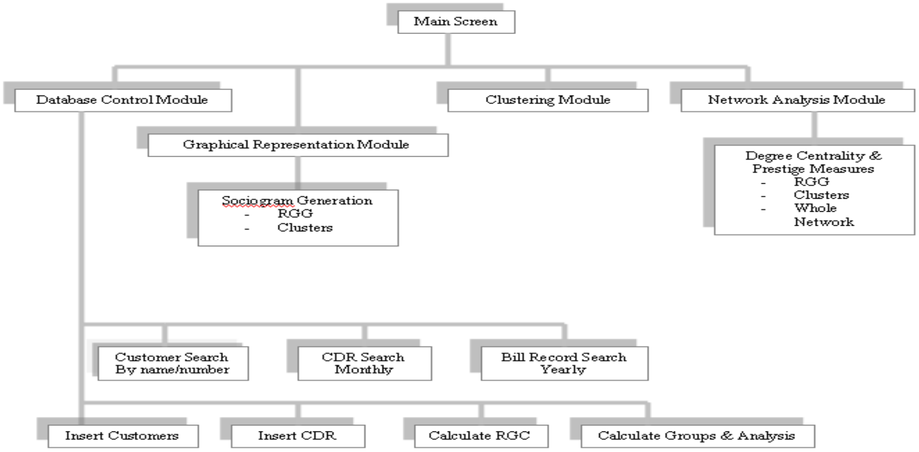


Figure 3.4. Functional View

3.3.1. Modules

This research based thesis requires a database which contains call detail records of the users in the mobile network. The database has already been discussed in detail in section 3.2. Now we need a Graphical User Interface also called front end application to provide various functionalities to the operator.

These functionalities includes

- Adding new customers.
- Adding new call detail record.
- Searching records of particular users from the database.
- Calculating Revenue generating customers for particular month of particular year.
- Calculating clusters or group members of any customer on the basis of summarized CDR maintained and the revenue generated by calling each member.
- Show the groups or clusters graphically calculated on above mentioned metrics.
- Make analysis of network which includes finding out the importance of each user in its group, cluster or in the whole network on the basis of centrality and prestige.

The basic modules for the system are as follows.

3.3.1.1. Database Controls Module

3.3.1.2. Graphical Representation Module

3.3.1.3. Network Analysis Module

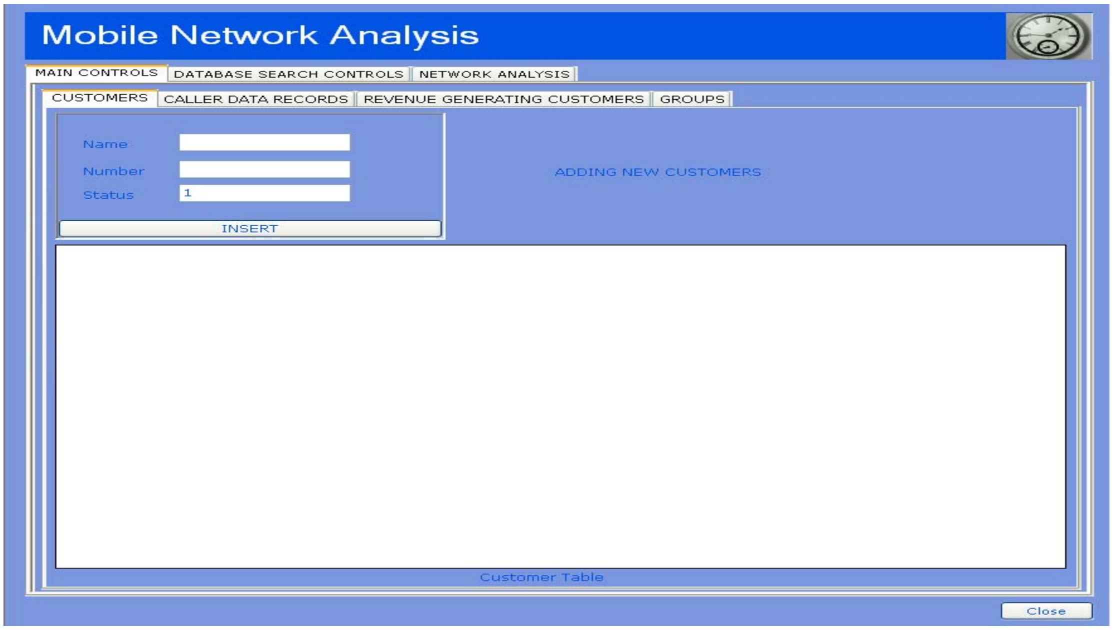
3.3.1.4. Clustering Module

3.3.1.1. Database Controls Module

The database controls includes search controls, main controls and links to group view and network analysis toolbox. All these controls reside on the main form of the application. The main from contains tree primary tab controls which further opens controls on the sub tabs.

3.3.1.1.1. Insert Customer

This sub tab contains controls for inserting new customer by specifying the name and number of customer. Further it displays the results in the data grid view and if there is any error the information will not be inserted and it will give operator a message box displaying the nature of error. It can be easily understood from figure 3.5



Name

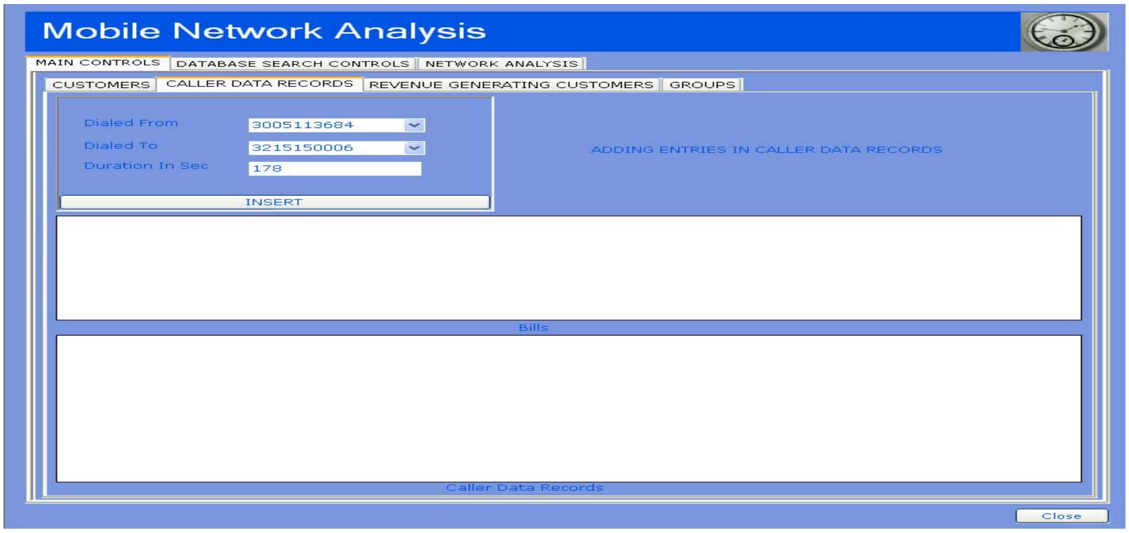
Number

Status 1

Fig 3.5 Adding new customer

3.3.1.1.2. Insert Call Detail Records

The controls in this sub tab can be used to insert new CDR by selecting the desired customers from the combo box list. Also it shows the updated bill of current month of the calling number and displays errors in the same way as discussed in section 3.2.1.1.1. Figure 3.6 describes the working of this control.



Dialed From 3005113684

Dialed To 3215150006

Duration In Sec 178

Fig 3.6 Adding new CDR

3.3.1.1.3. Calculate Revenue Generating Customers

The controls in this sub tab allow the operator to select specific month, year and lowest limit of revenue generated by any customer. It displays the bills of all customers and the revenue generating customers of selected inputs.

As explained earlier in database development that both triggers and coding in the front end application is used to implement this pseudo code and make the system faster and reliable.

3.3.1.1.4. Calculate Revenue Generating Groups

The controls in this sub tab allows the user to calculate the group members for the selected customer number and displays the closely connected user records maintained in database specific to only selected customer.

This sub tab also contains a button named “Launch Group View Tool” which provides a link to Cluster View Module.

3.3.1.2. Graphical Representation Module

This module shows the calculated group members for specified customer number, month, year and revenue per member. It has two methods for graphical representation. In both the method screen coordinates are selected so that all nodes or members appear in circular fashion i.e. nodes will be displayed at the boundary of circle.

Algorithm for generating points

Variables

x and **y** are the x and y coordinates

(**xc** , **yc**) is the center of circle

rad is the radius of circle

Dt index variable

Dpoints array for storing x , y coordinates

p is decision parameter for next points generation

sd is variable for introducing gaps between generated point at circumference of circle

Functions

draw8(x,y,xc,yc) function generates points at circumference of circle i.e 2 points in each quadrant

```
          // Inputs radius rad and circle centers x, y. The first point on
          //circumference of circle is (x,y) = (0,rad). Dt is used as index for point
          //array Dpoints
int x, y, p,Dt; x = 0; y = rad; Dt=0;
          // p the decision parameter for midpoint parameter
p = 1 - rad;
          // variables for introducing spaces between generated points
int sd = 0;
          // calculate points and stores in point array Dpoints
while (x < y)
{             sd++;
              x++;
          //Decision for next points. if p < 0 then next point along the circle is
          //(x+1, y) else its (x+1, y -1)
              if (p < 0)
               {    p += 2 * x + 1;
              }
              else
              {    y--;
                   p += 2 * (x - y) + 1;
              }
```

```
                if (sd == 59)
                {    //Stores the generated points Dp in the point array Dpoints at
                     //index Dt
                     draw8(x, y, xc, yc);
                     sd = 0;
                }
}
            //Dp is the generated point which is stored in the point array Dpoints at
            //index value Dt
void draw8(int x, int y, int xc, int yc)
{           Dp = new Point(xc + x, yc + y);
            Dpoints[Dt] = Dp; Dt++;
            Dp = new Point(xc - x, yc + y);
            Dpoints[Dt] = Dp; Dt++;
            Dp = new Point(xc + x, yc - y);
            Dpoints[Dt] = Dp; Dt++;
            Dp = new Point(xc - x, yc - y);
            Dpoints[Dt] = Dp; Dt++;
            Dp = new Point(xc + y, yc + x);
            Dpoints[Dt] = Dp; Dt++;
            Dp = new Point(xc - y, yc + x);
            Dpoints[Dt] = Dp; Dt++;
            Dp = new Point(xc + y, yc - x);
            Dpoints[Dt] = Dp; Dt++;
            Dp = new Point(xc - y, yc - x);
```

```
        Dpoints[Dt] = Dp; Dt++;
}
```

Algorithm for Drawing Links between nodes

Variables

U *is data table which stores the CDR records*

tmp *is array to stores the required list of customers*

Functions

get_CDR() *gets the required CDR's from database*

get_cust_ID(U) *gets the unique customers ID's from the CDR records stored in data table U*

```
        //Get the required CDR records and store in U table
U = get_CDR();
        //Get unique customer ID's and store in tmp array
tmp = get_cust_ID( U);
        // Displaying the customer ID's at generated points by getting values
        //from point array Dpoints in a way that customers ID at index 1 of tmp
        //array is allotted point at index 1 in Dpoints
for (int i = 0; i < tmp.elements.count ; i++)
{
    dc.Graphics.DrawString(tmp[i].ToString(),Dpoints[i]);
}
        //Drawing lines between nodes by reading records one by one from U
        //i.e if the customer ID initiating call is at index 3 in tmp and customer
```

```
          //ID receiving call is at index 7 in tmp table then draw line starting from
          //point at index 3 to point at index 7.
for (int j = 0; j < U.Rows.Count; j++)
{         dc.Graphics.DrawLine(RedPen,
          Dpoints[search(U.Rows[j].ItemArray[0].ToString(), k)],
          Dpoints[search(U.Rows[j].ItemArray[1].ToString(), k)]);
//The search function returns the index number of that customer ID in tmp
}
```

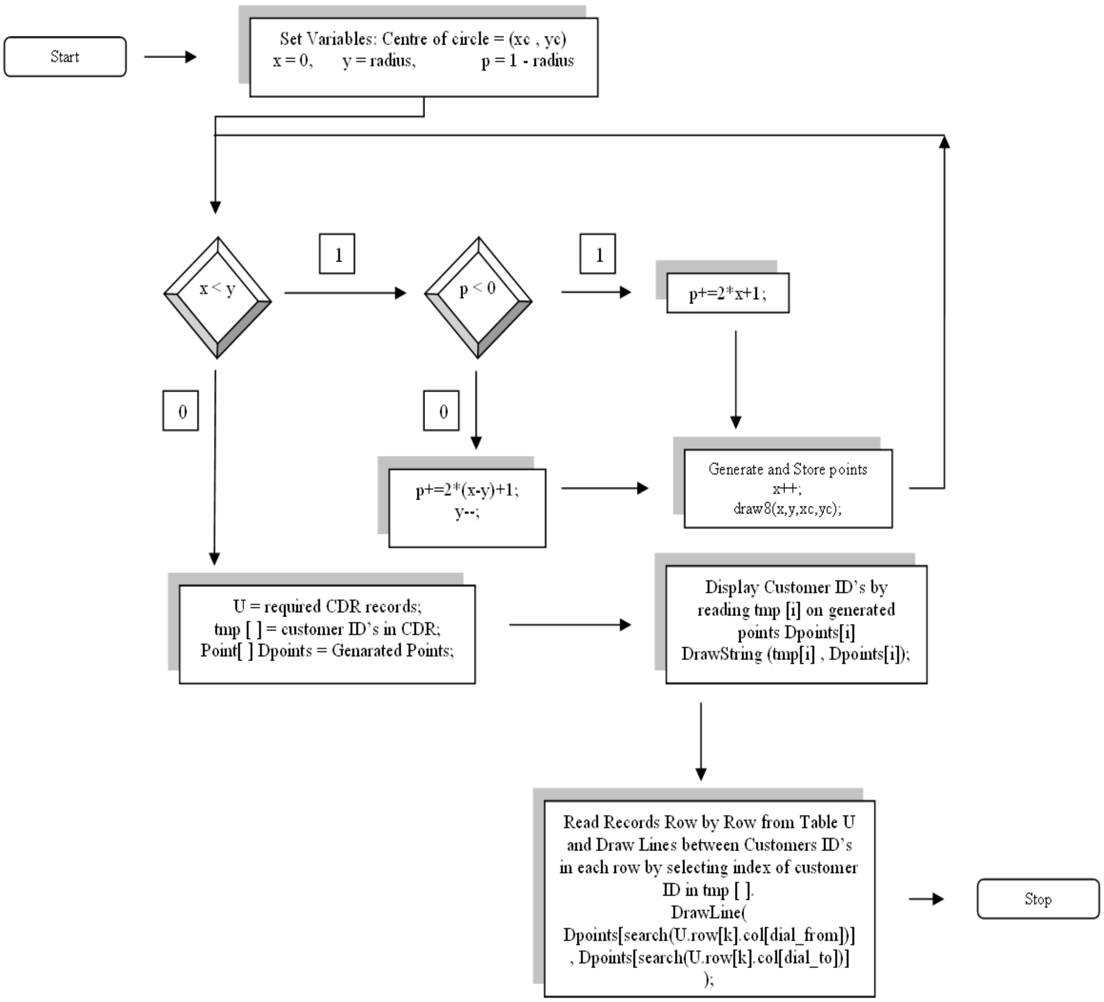

Start
Set Variables: Centre of circle = (xc , yc)
x = 0,   y = radius,   p = 1 - radius
x < y
1
p < 0
1
p+=2*x+1;
0
0
p+=2*(x-y)+1;
y--;
Generate and Store points
x++;
draw8(x,y,xc,yc);
U = required CDR records;
tmp [ ] = customer ID's in CDR;
Point[ ] Dpoints = Genarated Points;
Display Customer ID's by
reading tmp [i] on generated
points Dpoints[i]
DrawString (tmp[i] , Dpoints[i]);
Read Records Row by Row from Table U
and Draw Lines between Customers ID's
in each row by selecting index of customer
ID in tmp [ ].
DrawLine(
Dpoints[search(U.row[k].col[dial_from])]
, Dpoints[search(U.row[k].col[dial_to])]
);
Stop

Fig 3.7. Working of GR Algorithm

3.3.1.3. Network Analysis Module

This module analyzes the network on the basis of Degree Centrality and Degree Prestige. It has further two components. The first component shows the analysis results for clusters for specified customer.

Algorithm for Computing Network Metrics

Variables

i *is index variable*

custlist *is array which contains the required customer ID's*

cdrlist *contain the required CDR records*

N *is variable which stores total number of nodes i.e total degree*

indeg *and* ***outdeg*** *variables are used to calculate in-degree and out-degree of specific customer ID in custlist which is further stored in arrays* ***degcen[ ]*** *and* ***degpre[ ]***

Functions

Get_cust_list() *function gets the required customer ID from database*

Get_cdr_list(cust_ID) *function get incoming and outgoing CDR's from database for specified customers*

cust_list_elements_count() *function counts the number of elements in array custlist i.e total number of customers*

count_indeg(cust_ID) *and count_outdeg(cust_ID) functions counts the total incoming and outgoing call records from cdr_list[ ] for specified customer ID*

Step 1 *// set index to zero to compute for first customer ID in custlist*

```
            i = 0;
            //get the list of required customers ID
            custlist[ ] = Get_cust_list();
Step 2      // get CDR from database for customer ID at index i in custlist
            cdrlist = Get_cust_cdr(custlist[i]);
            // Calculate the maximum degree N
            N = cust_list_elements_count() - 1;
Step 3      //if CDR found calculate in-degree and out-degree from records
            //in cdrlist[ ] and stores in degcen and degpre arrays
            if (cdrlist [] != NULL )
            {       indeg = count_indegree(custlist[i]);
                    outdeg = count_outdegree(custlist[i];
                    degcen[i] = outdeg / N
                    degpre[i] = indeg / N
                    i++;
            }
            //if index is less than or equal to the maximum degree repeat the
            //procedure for rest of customers ID's
            if(i <= N)
            {
            goto Step2
            }
            else
            {
            Display Results       }
```

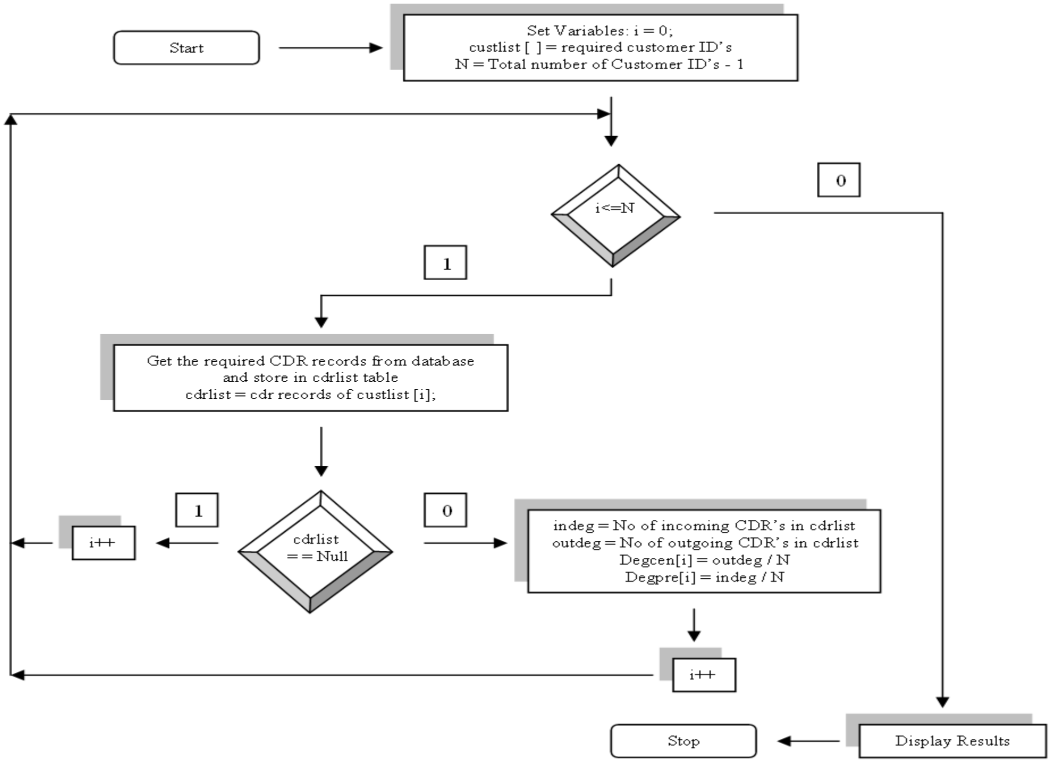

Start
Set Variables: i = 0;
custlist [ ] = required customer ID's
N = Total number of Customer ID's - 1
i<=N
0
1
Get the required CDR records from database
and store in cdrlist table
cdrlist = cdr records of custlist [i];
1
cdrlist
= = Null
0
i++
indeg = No of incoming CDR's in cdrlist
outdeg = No of outgoing CDR's in cdrlist
Degcen[i] = outdeg / N
Degpre[i] = indeg / N
i++
Stop
Display Results

Figure 3.8. Working of Network Metrics Algorithm

3.3.1.4. Clustering Module

In this module algorithm has been developed to make groups with in the mobile network for selected month and year on the basis of CDR. Algorithm basically reads the CDR and on the basis of incoming and outgoing call logs it assigns the connected users a particular group ID and does the same for all the other users in the network.

Data Structure has been maintained to keep the list of users belonging to the network. The data structure has some fields to keep the current status and group ID of added users. The fields are nodnam, group and status. These fields will be discussed later in this section in the pseudo code description.

Algorithm for computing clusters

The data structure can be defined as follows.

```
public struct nodes
   {
       public string nodnam;
       public int status;
       public int group;
   };
```

The nodnam field stores the customer ID which in this case is the phone number assigned to the customer. The status and group fields' values can be seen in table 3.1. Initially these fields are assigned as 0. X means other than 0.

Status	Group	Description
0	0	Not Processed
1	0	Not Possible
2	0	Processed and has no CDR
0	X	Not Possible
1	X	Under Process and group assigned
2	X	Processed and group assigned

Table 3.1 Fields Description

Variables

currentgroup *variable stores the recent assigned cluster ID and is incremented after discovering a cluster*

custlist *array stores the customer ID's of all customers obtained from database*

cdr *data table contains the CDR records of specific customer*

Functions

GetCustlist() *function gets the list of customer ID's from database*

FillDS_Custlist (int,int) *it fills the above mentioned data structure where nodnam contains customer ID's, and status and group parameters are assigned zero initially*

get_CDR(cust_ID) *function gets the CDR records of specified customer ID*

SET_STATUS(int) *function sets the status of nodnam for cur_cust_ID*

SET_STATUS_CDR_LIST(int) *function sets the status of all nodnam in the cdr of cur_cust_ID*

SET_GRP_CDRLIST(int) *function sets the group parameter in the data structure of all the nodnam present in current cdr*

search_nod(status,group) *function returns the nodnam with specified status and group*

```
Step1        currentgroup=1;
             custlist [ ] = GetCuslist();
             FillDS_Custlist(0,0); // status=0,group=0
Step 2       // select a nodnam having status and group assigned as zero
             cur_cust_ID = search_nod(0,0);
             // gets CDR records in cdr variable for cur_cust_ID
             cdr = get_CDR(cur_cust_ID);
             If (cdr == NULL)
             {       //sets the staus of cur_cust_ID in nodnam as 2
                     SET_STATUS(2);
                     repeat step 2
```

```
            }
            else
            {       //sets the status of cur_cust_ID in nodnam as 2 and set
                    //its group and the group of all nodnam in its cdr as
                    //currentgroup and set the status of all nodnam in its cdr
                    //as 1
                    SET_STATUS(2);
                    SET_STATUS_CDR_LIST(1);
                    SET_GRP_CDRLIST(currentgroup);
            }
Step 3      //Repeated until all the nodnam having
            //status =1 are processed.
            if (search_nod(1,currentgroup ) != NULL )
            {       goto Step 2.
            }
Step 4      //if there is still any nodnam whose status and group is zero than
            //repeat the same procedure for that nodnam
            if (search_nod(0,0) != NULL )
            {       cur_cust_ID = search_nod(0,0);
                    currentgroup++;
                    Goto step2;
            }
            else
            {       Display results and quit algorithm
            }
```

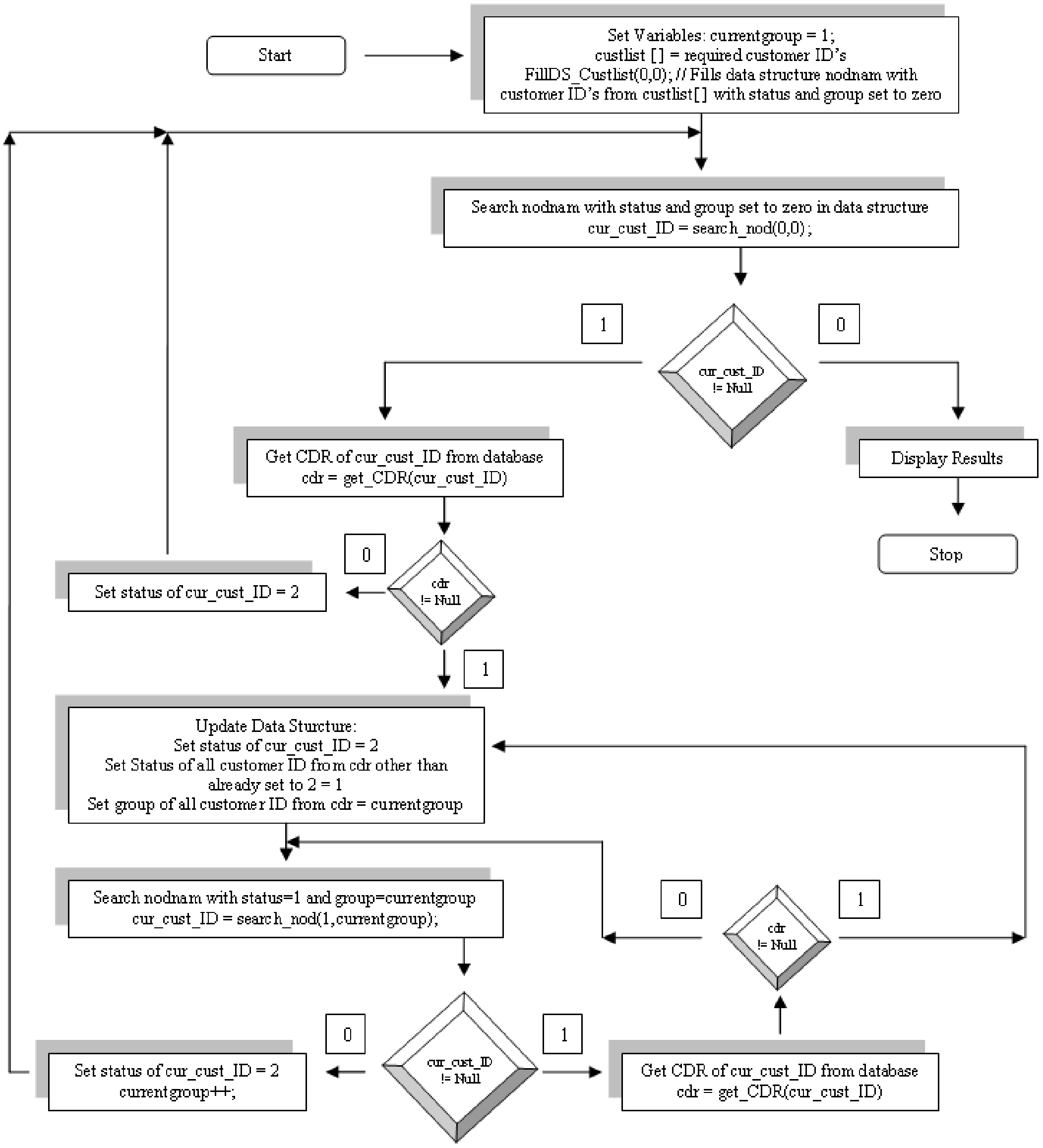

Start
Set Variables: currentgroup = 1;
custlist [] = required customer ID's
FillDS_Custlist(0,0); // Fills data structure nodnam with
customer ID's from custlist[] with status and group set to zero
Search nodnam with status and group set to zero in data structure
cur_cust_ID = search_nod(0,0);
1
cur_cust_ID
!= Null
0
Get CDR of cur_cust_ID from database
cdr = get_CDR(cur_cust_ID)
Display Results
Stop
0
Set status of cur_cust_ID = 2
cdr
!= Null
1
Update Data Sturcture:
Set status of cur_cust_ID = 2
Set Status of all customer ID from cdr other than
already set to 2 = 1
Set group of all customer ID from cdr = currentgroup
Search nodnam with status=1 and group=currentgroup
cur_cust_ID = search_nod(1,currentgroup);
0
cdr
!= Null
1
0
Set status of cur_cust_ID = 2
currentgroup++;
cur_cust_ID
!= Null
1
Get CDR of cur_cust_ID from database
cdr = get_CDR(cur_cust_ID)

Figure 3.9. Working of Clustering Algo

Chapter 4

Results and Analysis

In this chapter we will consider an example of network and we will make our analysis and will review the results of the analysis. For simplicity reasons we will consider 100 users and we will generate CDR for these users on the basis of which analysis will be made.

4.1. Data Generation for CDR Analysis

First of all we will generate the list of customers by using database control modules. The generated list is shown in the following table.

3215150000	3215150020	3215150040	3215150060	3215150080
3215150001	3215150021	3215150041	3215150061	3215150081
3215150002	3215150022	3215150042	3215150062	3215150082
3215150003	3215150023	3215150043	3215150063	3215150083
3215150004	3215150024	3215150044	3215150064	3215150084
3215150005	3215150025	3215150045	3215150065	3215150085
3215150006	3215150026	3215150046	3215150066	3215150086
3215150007	3215150027	3215150047	3215150067	3215150087
3215150008	3215150028	3215150048	3215150068	3215150088
3215150009	3215150029	3215150049	3215150069	3215150089
3215150010	3215150030	3215150050	3215150070	3215150090
3215150011	3215150031	3215150051	3215150071	3215150091
3215150012	3215150032	3215150052	3215150072	3215150092
3215150013	3215150033	3215150053	3215150073	3215150093
3215150014	3215150034	3215150054	3215150074	3215150094
3215150015	3215150035	3215150055	3215150075	3215150095
3215150016	3215150036	3215150056	3215150076	3215150096
3215150017	3215150037	3215150057	3215150077	3215150097
3215150018	3215150038	3215150058	3215150078	3215150098
3215150019	3215150039	3215150059	3215150079	3215150099

Table 4.1. List of generated Users

After generating the list of user, 438 call data records have been generated randomly. The call data records are inserted in CDR table in the database. For each entry in CDR, bill table will be updated. Also the caller group table will also be updated which will keep the summary of CDR table that would be used by the algorithms in the different modules so that the system should work faster. The example of CDR, Bill and caller group table is shown below.

These are the entries for user 3215152002 in CDR Table.

Dial from	Dial to	Duration	Cost
3215150002	3215150011	180	9.00
3215150002	3215150011	180	9.00
3215150002	3215150020	125	6.25
3215150002	3215150020	125	6.25
3215150002	3215150020	125	6.25
3215150002	3215150020	125	6.25
3215150002	3215150089	60	3.00
3215150002	3215150089	60	3.00
3215150002	3215150089	60	3.00

Table 4.2. CDR Example

The Bill Table will have the updated bill and will have only one entry for the user.

User	Total Bill	Date
3215150002	52.0000	2007-09-12 14:28:00.000

Table 4.3. Bill Example

The caller group Table will contain the summary of the call data records for the user which are

Dial From	Dial to	Total Revenue
3215150002	3215150011	18.0000

3215150002	3215150020	25.0000
3215150002	3215150089	9.0000

Table 4.4. Summarized CDR Example

Notice that there are only three entries in Caller group Table for nine entries in the CDR table for this user. So for calculating group and in auto clustering we have to make fewer computations by using caller group table instead of CDR table. Please note that all these entries have been made for the month September 2007.

4.2. Identification and Analysis of Valued Customers

The valued customers can be identified by the following methods as discussed earlier in the objectives.

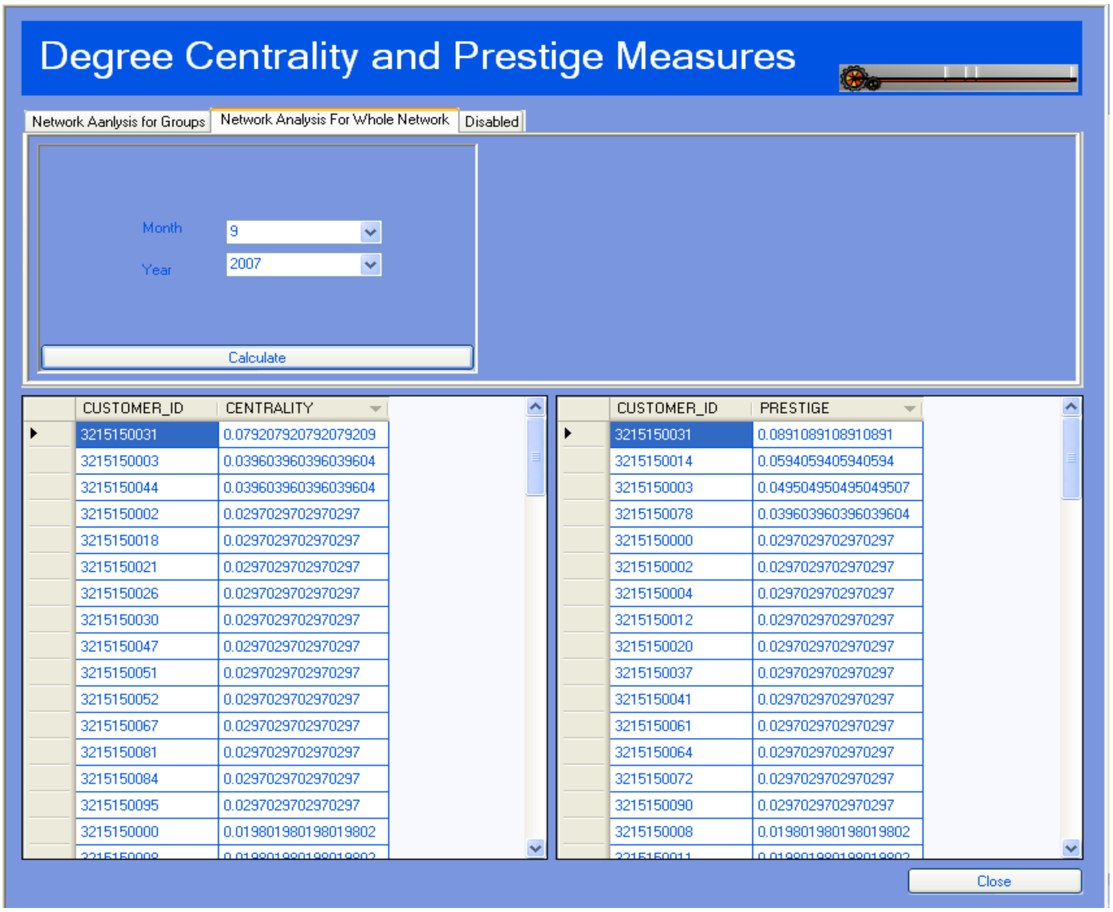
4.1.1.1. Network Analysis for whole network

4.1.1.2. Revenue Generating Customers

4.1.1.3. Revenue Generating Group and its Network Analysis

4.2.1. Network Analysis for whole network

The network analysis has been made on degree centrality and prestige basis. The degree centrality metric identifies the extent to which the user has out-links with others i.e. the higher this factor the more number of users he is calling. The degree prestige metric identifies the extent to which the user has in-links with others i.e. the higher this factor the more number of users are calling him. Figure 4.1 shows the network metrics for all the generated users as computed from CDR.



Degree Centrality and Prestige Measures

Network Aanlysis for Groups
Network Analysis For Whole Network
Disabled

Month: 9

Year: 2007

CUSTOMER_ID	CENTRALITY
3215150031	0.079207920792079209
3215150003	0.039603960396039604
3215150044	0.039603960396039604
3215150002	0.0297029702970297
3215150018	0.0297029702970297
3215150021	0.0297029702970297
3215150026	0.0297029702970297
3215150030	0.0297029702970297
3215150047	0.0297029702970297
3215150051	0.0297029702970297
3215150052	0.0297029702970297
3215150067	0.0297029702970297
3215150081	0.0297029702970297
3215150084	0.0297029702970297
3215150095	0.0297029702970297
3215150000	0.019801980198019802
3215150008	0.019801980198019802

CUSTOMER_ID	PRESTIGE
3215150031	0.0891089108910891
3215150014	0.0594059405940594
3215150003	0.049504950495049507
3215150078	0.039603960396039604
3215150000	0.0297029702970297
3215150002	0.0297029702970297
3215150004	0.0297029702970297
3215150012	0.0297029702970297
3215150020	0.0297029702970297
3215150037	0.0297029702970297
3215150041	0.0297029702970297
3215150061	0.0297029702970297
3215150064	0.0297029702970297
3215150072	0.0297029702970297
3215150090	0.0297029702970297
3215150008	0.019801980198019802
3215150011	0.019801980198019802

Figure 4.1. Network Metrics for whole network

Notice that 3215150031 has highest degree centrality as well as degree prestige in all the users. It can be deduced from this result that 3215150031 might be the revenue generating customer in two ways i.e. by calling other people he is generating revenue and also when other people calls him he become the source for revenue generation. These metrics are based on the number of in-links and out-links so it cannot be deduces from these results only that a customer is a revenue generating customer. The results of this module combined with results of RGC and RGG can confirm whether a customer is generating revenue generating customer or not. Furthermore if a customer has zero degree centrality and degree prestige he might have switched to some other network. The following chart shows the results of network metrics for first ten users.

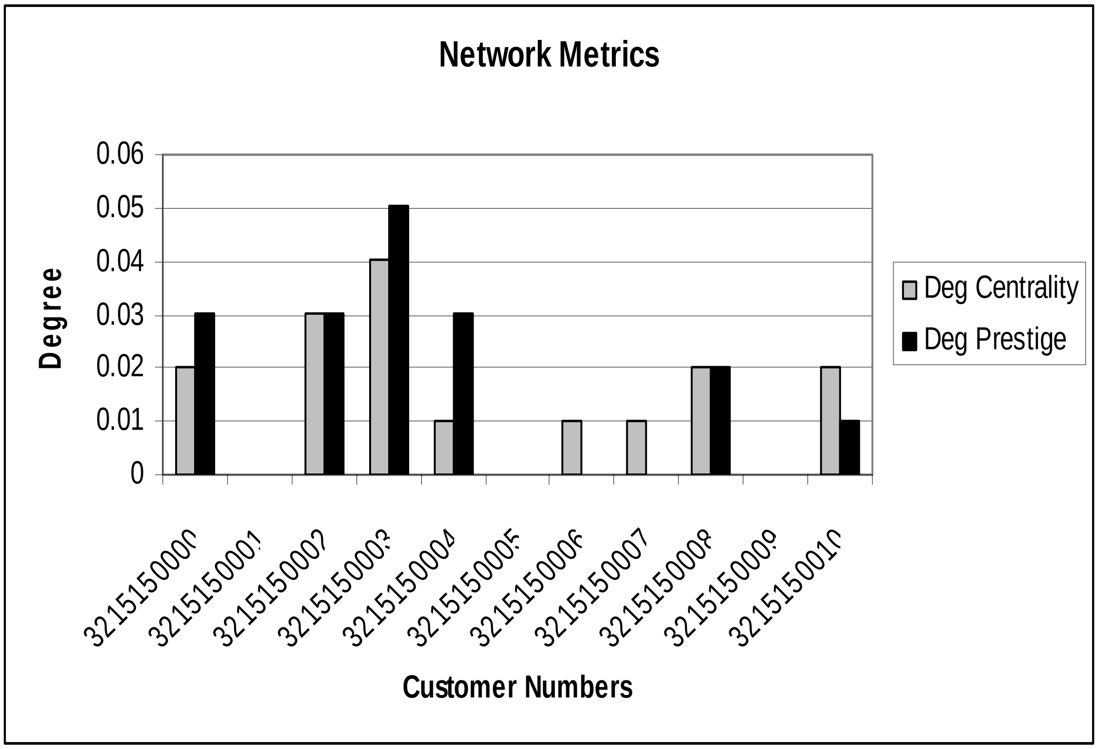


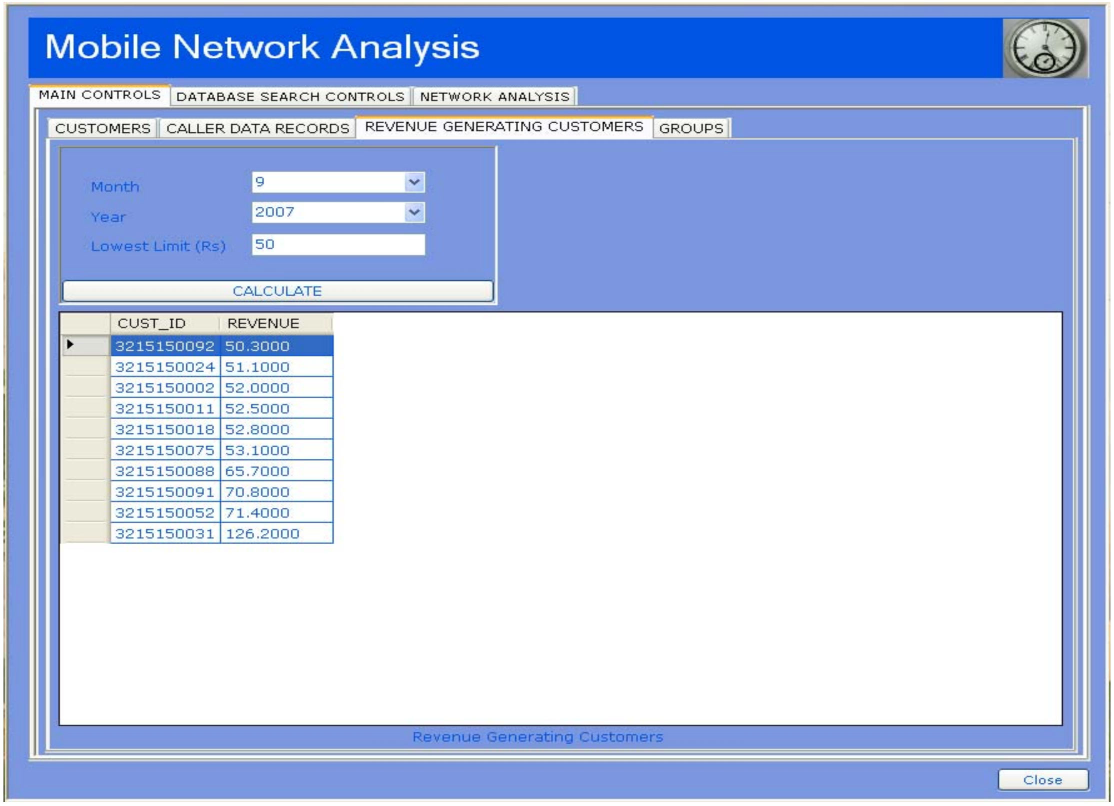
Figure 4.2. Network Metrics Chart

It can be seen from above chart that both the network metrics for user 3215150001, 3215150005 and 3215150009 are zero indicating their inactiveness in the network.

4.2.2. Revenue Generating Customers

Revenue generating customers are the key or valued customers for the company as they generate higher revenue for the company. The company would be keen to know about their valued customers so that if they introduce any new package they could offer them to their valued customers first.

The method to find out revenue generating customers has been made simple by using bills table. If we use CDR table to calculate revenue generating customers first we have to sum the cost of all the outgoing call records for all the users one by one and then we can find out the desired results up to the criteria. The problem with this approach is that we have to make lots of computations every time which will make the system slow so to use the bill table is feasible as it already has the updated bills of all the customers which has been made possible by triggers in CDR Table. The following figure shows the actual working of the system.



Month: 9

Year: 2007

Lowest Limit (Rs): 50

CUST_ID	REVENUE
3215150092	50.3000
3215150024	51.1000
3215150002	52.0000
3215150011	52.5000
3215150018	52.8000
3215150075	53.1000
3215150088	65.7000
3215150091	70.8000
3215150052	71.4000
3215150031	126.2000

Figure 4.3. Revenue Generating Customer Example

In figure 4.3 the lowest revenue limit has been set to 50. Suppose 50 means 50*100 = 5000 and this limit 5000 is the criteria for finding RGC i.e. any customer generating revenue greater than or equal to 5000 is a revenue generating customer then it can be easily seen from fig 4.3 that out of 100 generated users only ten users meet the criteria. The following chart shows the revenue generated by all the customers in the specified month.

Figure 4.4 shows the revenue generated by all the customers in the month of September. It can be confirmed that the results computed in figure 4.3 are absolutely fine as there are only 10 bars touching or crossing the limit of 50.

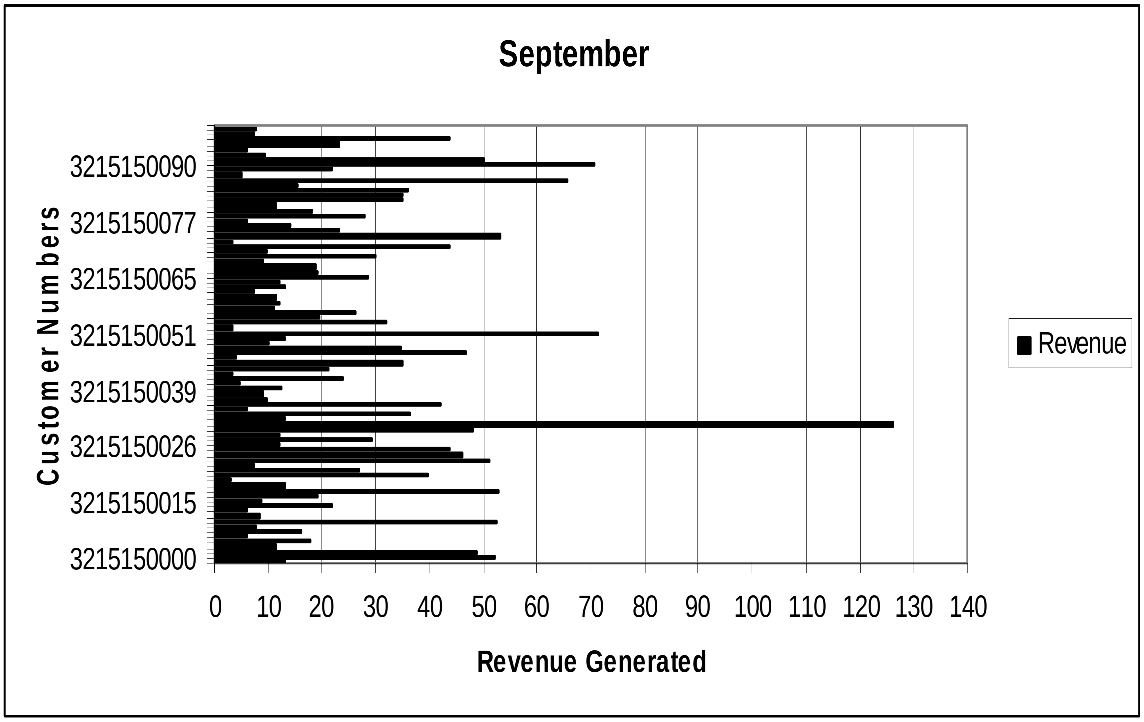
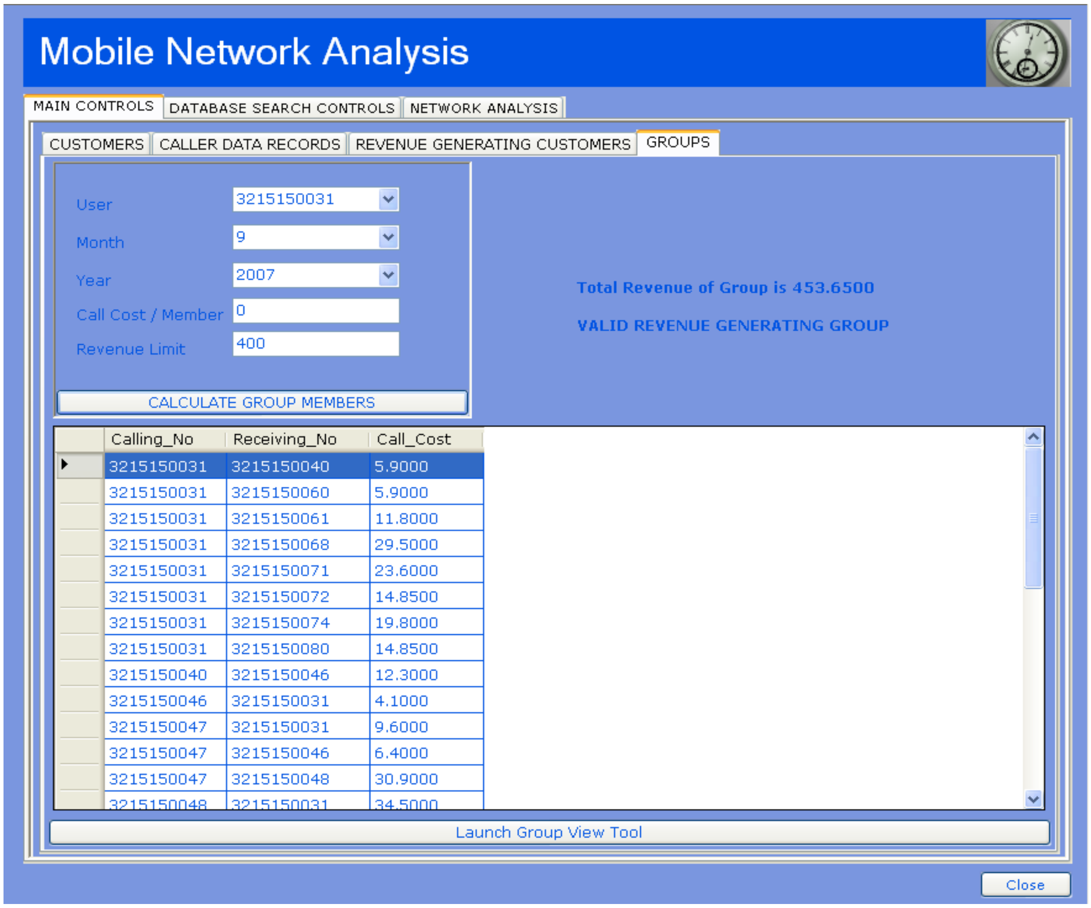


Figure 4.4. Bills of All customers

4.2.3. Revenue generating group and Analysis

We can also find out the closely connected members in the network for particular customer. RGG module can be used to find out the person closely connected to any person e.g. In above section if the company want to introduce its new package not only to revenue generating customers but also to the users closely related to the revenue generating customer.



User	3215150031
Month	9
Year	2007
Call Cost / Member	0
Revenue Limit	400

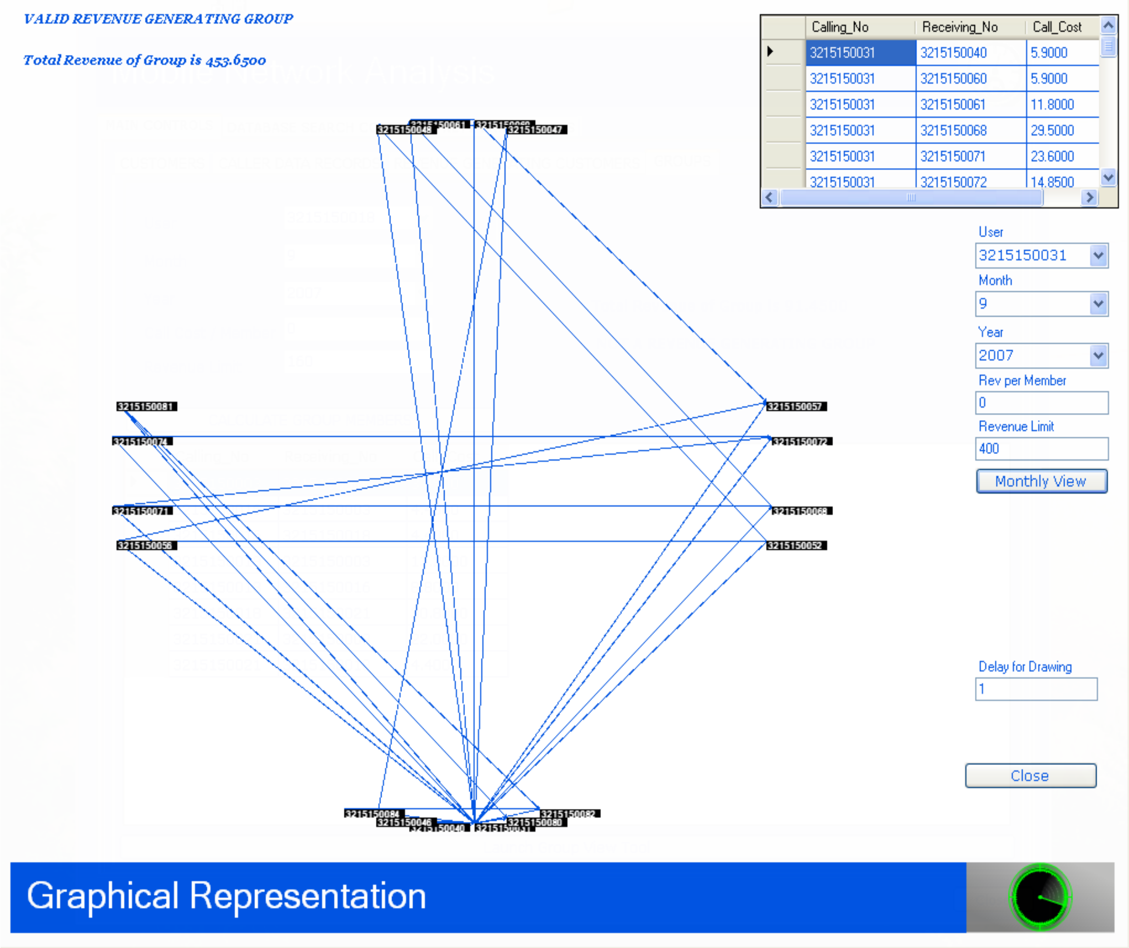
Total Revenue of Group is 453.6500

VALID REVENUE GENERATING GROUP

Calling_No	Receiving_No	Call_Cost
3215150031	3215150040	5.9000
3215150031	3215150060	5.9000
3215150031	3215150061	11.8000
3215150031	3215150068	29.5000
3215150031	3215150071	23.6000
3215150031	3215150072	14.8500
3215150031	3215150074	19.8000
3215150031	3215150080	14.8500
3215150040	3215150046	12.3000
3215150046	3215150031	4.1000
3215150047	3215150031	9.6000
3215150047	3215150046	6.4000
3215150047	3215150048	30.9000
3215150048	3215150031	34.5000

Figure 4.5 Revenue Generating Group

In section 4.2.2 we found out the RGC and 3215150031 was generating the maximum revenue i.e. 12623. It RGG consists of all the members calling this customers or receiving calls from this customer. All the CDRs generated by these members by calling in this group are used to compute the total revenue generated by the group. We can also filter the members by specifying the call cost / member. The following figure shows the sociogram for the specified number obtained from the GR module.



VALID REVENUE GENERATING GROUP

Total Revenue of Group is 453.6500

Calling_No	Receiving_No	Call_Cost
3215150031	3215150040	5.9000
3215150031	3215150060	5.9000
3215150031	3215150061	11.8000
3215150031	3215150068	29.5000
3215150031	3215150071	23.6000
3215150031	3215150072	14.8500

User
3215150031

Month
9

Year
2007

Rev per Member
0

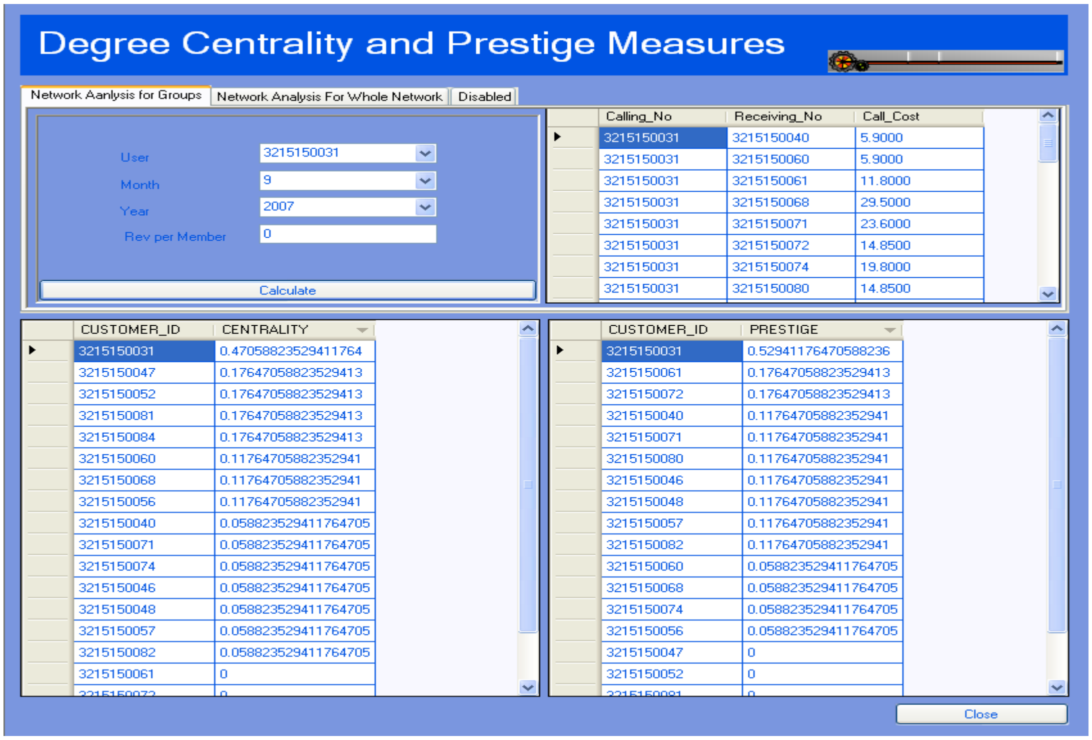
Revenue Limit
400

Delay for Drawing
1

Figure 4.6 Sociogram Example

Its clear from figure 4.6 to how many users the customer 3215150031 is linked to and how much revenue is being generated by the call between these users.

Valued customers can also be discovered by network analysis module. The network analysis module finds out the degree centrality and degree prestige of the users in the network. Higher the degree centrality means higher number of calls has been made by that user and higher the degree prestige higher number of calls has been received by the user. Both the users will be the most important users in the whole network. The one with higher centrality can be the user generating handsome revenue. And the one with higher prestige is also responsible for revenue generation not by itself but by the users who are calling that customer.


Network Aanlysis for Groups
Network Analysis For Whole Network
Disabled
Calculate
Close

Degree Centrality and Prestige Measures

User 3215150031
Month 9
Year 2007
Rev per Member 0

Calling_No	Receiving_No	Call_Cost
3215150031	3215150040	5.9000
3215150031	3215150060	5.9000
3215150031	3215150061	11.8000
3215150031	3215150068	29.5000
3215150031	3215150071	23.6000
3215150031	3215150072	14.8500
3215150031	3215150074	19.8000
3215150031	3215150080	14.8500

CUSTOMER_ID	CENTRALITY
3215150031	0.47058823529411764
3215150047	0.17647058823529413
3215150052	0.17647058823529413
3215150081	0.17647058823529413
3215150084	0.17647058823529413
3215150060	0.11764705882352941
3215150068	0.11764705882352941
3215150056	0.11764705882352941
3215150040	0.058823529411764705
3215150071	0.058823529411764705
3215150074	0.058823529411764705
3215150046	0.058823529411764705
3215150048	0.058823529411764705
3215150057	0.058823529411764705
3215150082	0.058823529411764705
3215150061	0
3215150072	0

CUSTOMER_ID	PRESTIGE
3215150031	0.52941176470588236
3215150061	0.17647058823529413
3215150072	0.17647058823529413
3215150040	0.11764705882352941
3215150071	0.11764705882352941
3215150080	0.11764705882352941
3215150046	0.11764705882352941
3215150048	0.11764705882352941
3215150057	0.11764705882352941
3215150082	0.11764705882352941
3215150060	0.058823529411764705
3215150068	0.058823529411764705
3215150074	0.058823529411764705
3215150056	0.058823529411764705
3215150047	0
3215150052	0
3215150081	0

Figure 4.7. Network Analysis for RGG

It can be seen from figure 4.1 that 3215150031 is the most important user in the network both with respect to centrality and prestige and it was also the revenue generating customer. The second most important user is 3215150003 with respect to centrality and 3215150014 with respect to prestige. Although both of them were not the revenue generating customers but their group might be revenue generating group. So they can also be the candidates for the new exciting offers introduced by the company.

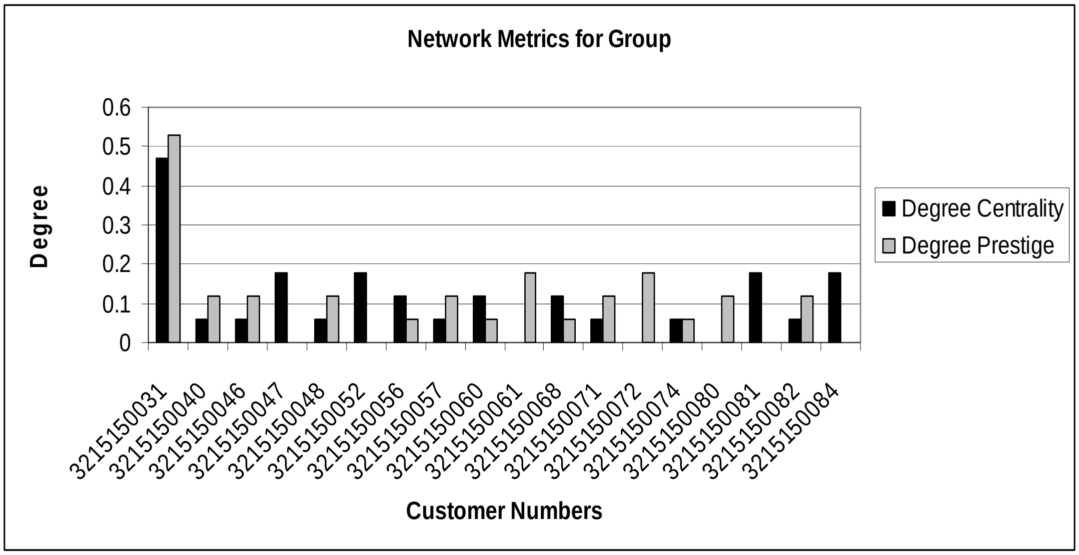
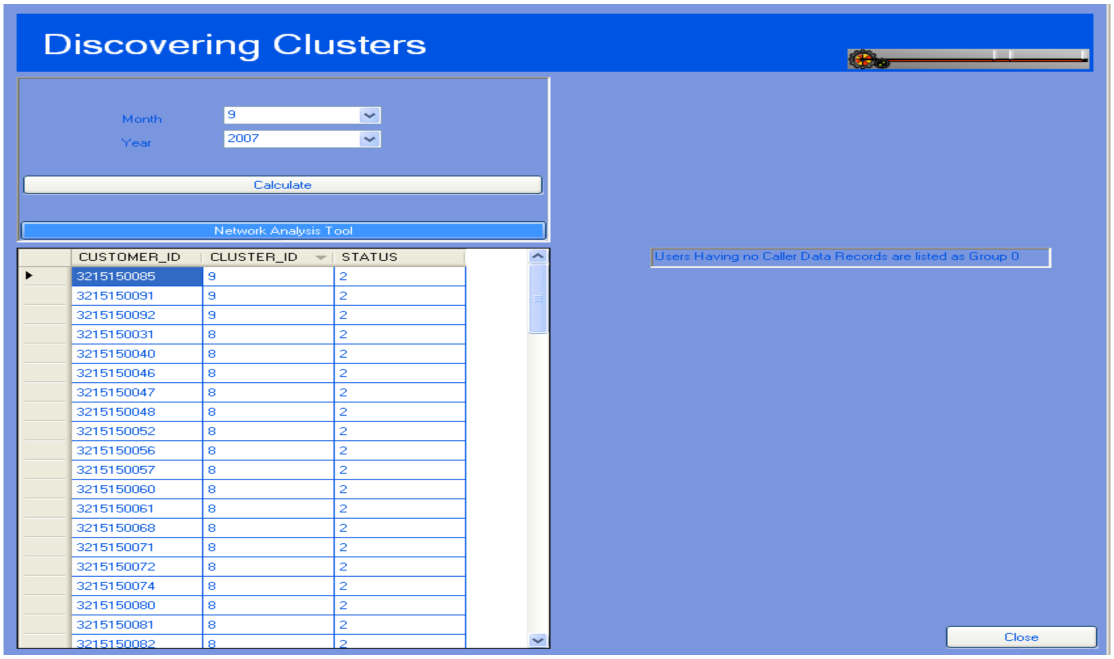


Figure 4.8. Network Metrics for RGG

The network analysis for the group has been shown in figure 4.8. It is obvious that 3215150031 is most important user w.r.t prestige in its group and also w.r.t centrality. So we can find out the most important users with in a group also. The second most important person w.r.t degree centrality is 3215150047 but w.r.t. degree prestige its 3215150061. It means that 3213130047 is generating revenue in its group by calling the members as its degree prestige is zero and 3215150061 is becoming a source of revenue generation b receiving calls from other members as its degree centrality is zero.

4.3. Identification and Analysis of Clusters

This module makes the clusters of users in the whole network and assigns them a particular Cluster ID. In section 4.2.2 we have calculated the group of users but those groups were only showing the closely connected user for the particular customer. But in auto clustering all the users connected directly or indirectly in the network are kept in one cluster thus making sub networks in one network.



Discovering Clusters

Month: 9
Year: 2007

CUSTOMER_ID	CLUSTER_ID	STATUS
3215150085	9	2
3215150091	9	2
3215150092	9	2
3215150031	8	2
3215150040	8	2
3215150046	8	2
3215150047	8	2
3215150048	8	2
3215150052	8	2
3215150056	8	2
3215150057	8	2
3215150060	8	2
3215150061	8	2
3215150068	8	2
3215150071	8	2
3215150072	8	2
3215150074	8	2
3215150080	8	2
3215150081	8	2
3215150082	8	2

Users Having no Caller Data Records are listed as Group 0

Figure 4.9. Clustering Module

The working of this module is already discussed in section 3.3.1.4. Figure 4.9 shows the clusters identified by filtering information from CDR. The full information of clusters can be seen in the following figure.

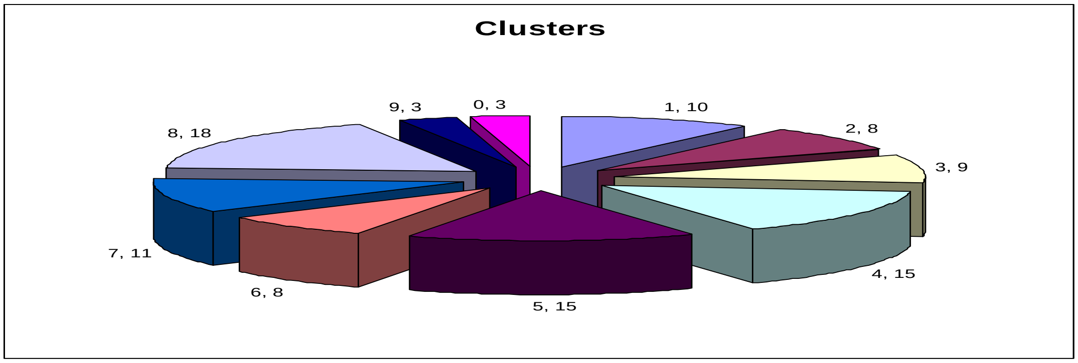
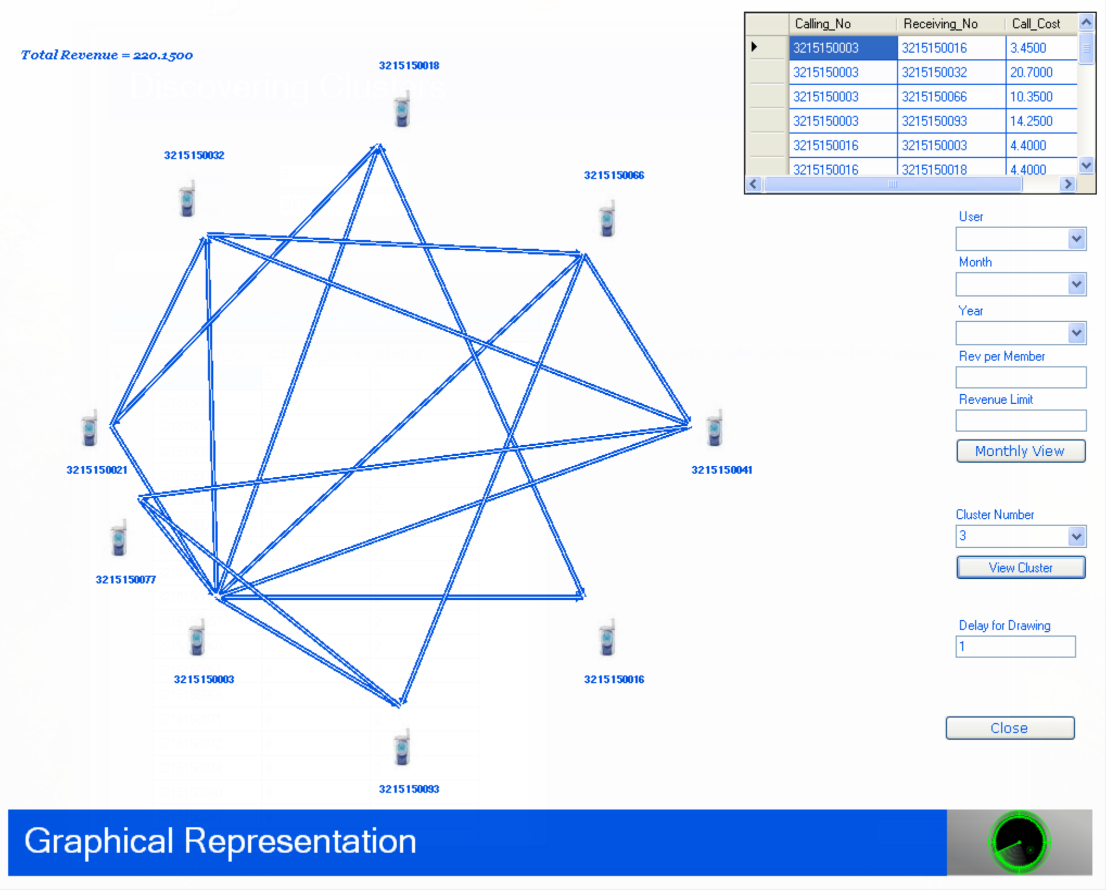


Figure 4.10. Clusters

Figure 4.9 shows the cluster ID and number of users in the cluster as discovered from the clustering module for the generated data. The value 9, 3 in this figure indicates that its cluster ID is 9 and number of users in this cluster are 3.

The generated sociogram for one of the computed cluster is as follows.



Total Revenue = 220.1500

Calling_No	Receiving_No	Call_Cost
3215150003	3215150016	3.4500
3215150003	3215150032	20.7000
3215150003	3215150066	10.3500
3215150003	3215150093	14.2500
3215150016	3215150003	4.4000
3215150016	3215150018	4.4000

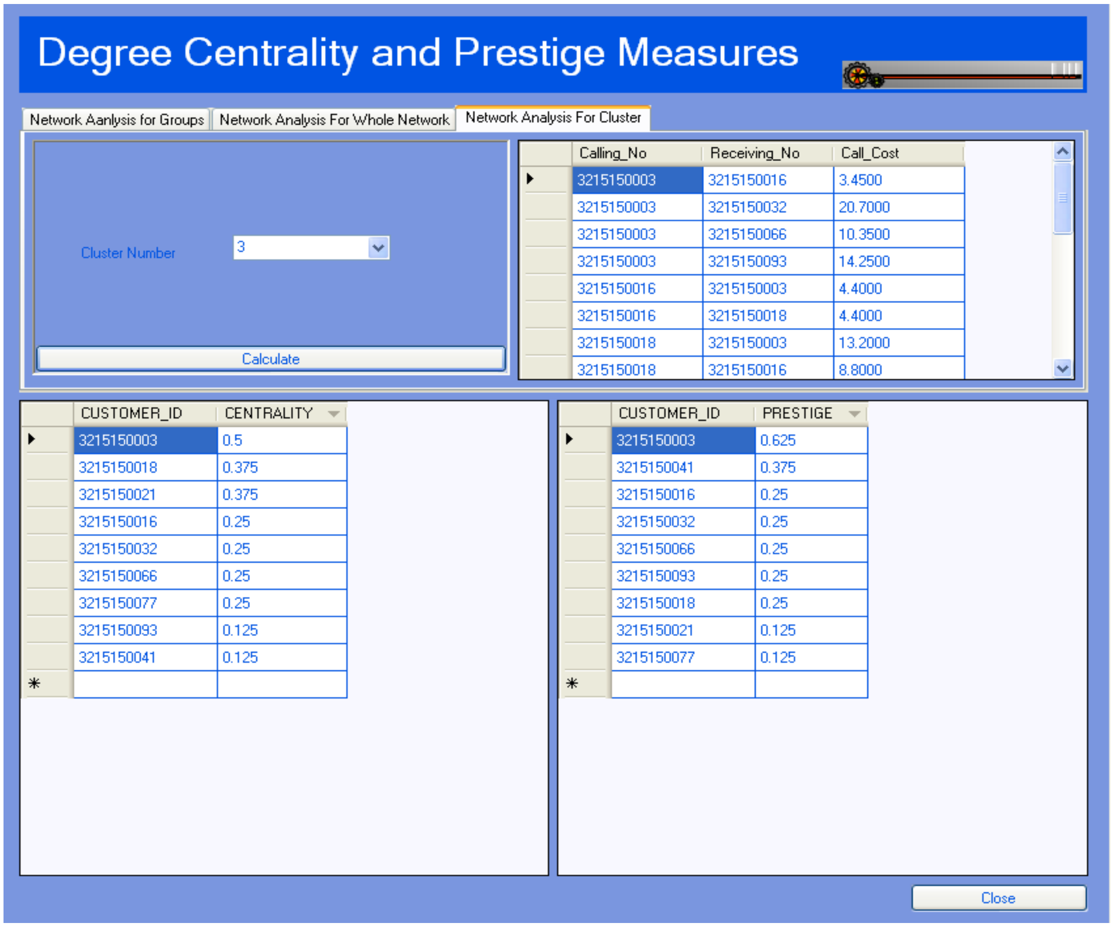
Cluster Number: 3

Delay for Drawing: 1

Figure 4.11. Sociogram for Cluster

This sociogram has been made for cluster number 3 which has nine users so confirming the results in figure 4.9.

Furthermore network analysis can also be made for specified cluster to find the prominence of users participating in that cluster with respect to its degree prestige or degree centrality as shown in the following figure.


Degree Centrality and Prestige Measures
Network Aanlysis for Groups
Network Analysis For Whole Network
Network Analysis For Cluster
Calculate
Close

Cluster Number: 3

Calling_No	Receiving_No	Call_Cost
3215150003	3215150016	3.4500
3215150003	3215150032	20.7000
3215150003	3215150066	10.3500
3215150003	3215150093	14.2500
3215150016	3215150003	4.4000
3215150016	3215150018	4.4000
3215150018	3215150003	13.2000
3215150018	3215150016	8.8000

CUSTOMER_ID	CENTRALITY
3215150003	0.5
3215150018	0.375
3215150021	0.375
3215150016	0.25
3215150032	0.25
3215150066	0.25
3215150077	0.25
3215150093	0.125
3215150041	0.125

CUSTOMER_ID	PRESTIGE
3215150003	0.625
3215150041	0.375
3215150016	0.25
3215150032	0.25
3215150066	0.25
3215150093	0.25
3215150018	0.25
3215150021	0.125
3215150077	0.125

Figure 4.12. Network Metrics for Clusters

The most prominent user as seen from the calculation in above figure is 3215150003 as it has the highest degree for both the centrality and prestige. It can be predicted from this result that he can be the leader of the communication network as discovered in this cluster. The following figure shows the chart for network metrics computation for this cluster.

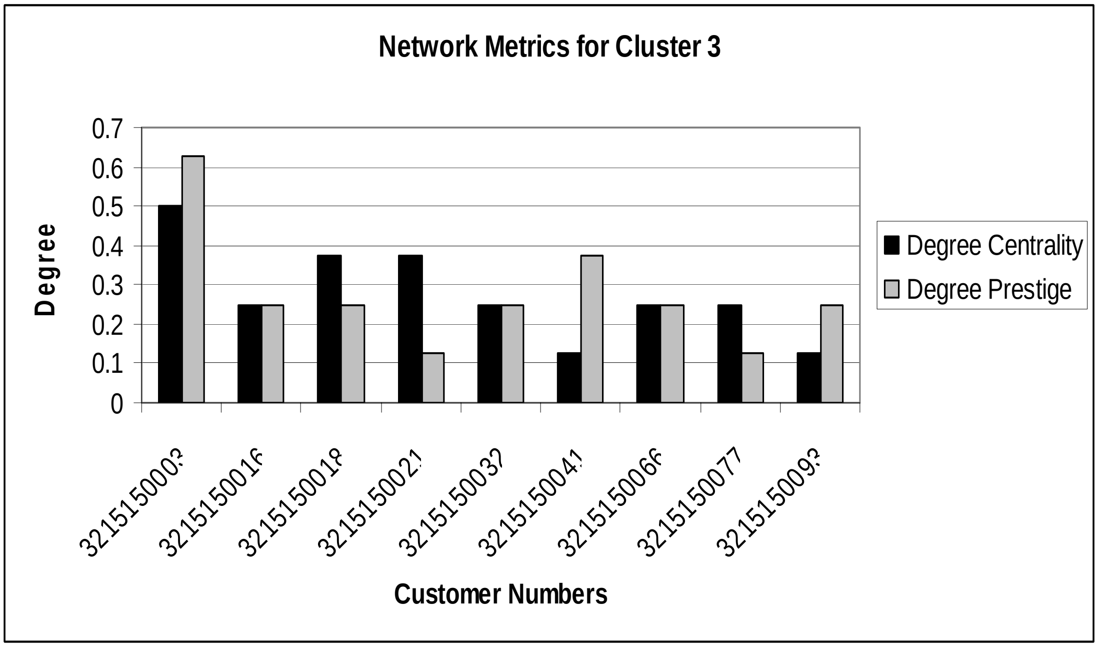


Figure 4.13. Network Metrics Chart for Cluster

Notice that when there is no customer having both the metrics as zero for clusters as well as for RGG which shows that every user in cluster or group is an active user as all the inactive users are kept in cluster number 0.

All these results shows that call detail records are enough for the identification and analysis of social network. The results of this analysis can be useful for finding the prominent users, closely connected users for any users and the complete network structure by discovering the clusters.

Chapter 5

Conclusions

5.1. Overview

Every time a call is placed on a telecommunications network, descriptive information about the call is saved as a call detail record. The number of call detail records that are generated and stored is huge. For example, AT&T long distance customers alone generate over 300 million call detail records per day (Cortes & Pregibon, 2001). Given that several months of call detail data is typically kept online, this means that tens of billions of call detail records will need to be stored at any time.

Call detail records include sufficient information to describe the important characteristics of each call. At a minimum, each call detail record will include the originating and terminating phone numbers, the date and time of the call and the duration of the call. Call detail records are generated in realtime and therefore will be available almost immediately for data mining. This can be contrasted with billing data, which is typically made available only once per month.

Call detail records can be used directly for social network analysis, since the goal of SNA to extract knowledge at the link level. Thus, the call detail records associated with a customer must be summarized into a single record that describes the customer’s calling behavior. The choice of summary variables (i.e., features) is critical in order to obtain a useful description of the customer. Below is a list of features that one might use when generating a summary description of a customer based on the calls they originate and receive over some time period.

1. Average call duration
2. Revenue generated by some call
3. Revenue generated by calling some customer
4. In-degree of some customer
5. Out-degree of some customer

These features can be used to build a customer profile. Such a profile has many potential applications. For example, it could be used to distinguish between active and inactive customers based on the in-degree and out-degree or the revenue generated by making or receiving calls. Most of the these features listed above were generated in a straightforward manner from the underlying data, but some features, such as the last two features, required a little more thought and creativity. Because most people call only a few number of people over a reasonably short period of time (e.g., a month), but there in-degree can specify how important they are in generating revenue by receiving calls from others.

Telecommunication companies, like other large businesses, may have millions of customers. By necessity this means maintaining a database of information on these customers. This information will include name and address information and may include other information such as service plan and contract information, credit score, family income and payment history. This information may be supplemented with data from external sources, such as from credit reporting agencies. Because the customer data maintained by telecommunication companies does not substantially differ from that maintained in most other industries, the social network analysis described in this research do not focus on this source of data. However, customer data is often used in conjunction with other data in order to improve results. For example, customer data is typically used to supplement call detail data when trying to identify phone fraud or network of some customer.

5.1. Areas of Applications

We have been through a comprehensive review of the social network analysis for call detail records which made a basis for our research. The results indicates that CDR provides a rich source for extracting social network information from mobile communication across network which is quite enough for discovering and analyzing social networks and prominence of an actor in the network as they do not only contain the billing information but also the linkage information of the communicating users.

The results can be used by Telecom Company or law enforcing agencies to filter out the information of their point of interest. The Telecom Company can use this system to find valued customers by calculating Revenue Generating Customer, Revenue Generating Group or by finding the prominence of some users by network analysis modules so that they could offer new exciting offers to their valued clients.

Furthermore the law enforcing agencies can use the information obtained by discovering clusters to track a network of some criminal and by network analysis they can find out the prominent member in that cluster not only on the basis of in-degree but also by comparing the out-degree of the users found in the cluster.

5.2. Future Work

In future this system can be enhanced by integrating it with GIS application to find out the exact location of some user as this information is also stored in CDR. Correlation can be done for the monthly calculated clusters to find out the switching of users to other operators or numbers. Also social networks can also be identified by using SMS logs, which can be correlated with our results to find out the missing members in the network who are just using SMS service to communicate with others.

List of Abbreviations

CDR – Call Detail Records

Data flow diagram (DFD) - a modeling notation that represents a functional decomposition of a system

DB – data base

RGC – Revenue Generating Customers

RGG – Revenue Generating Group

GIS- Graphical Information System

GUI – Graphical User Interface

SCDR – Summarized Call Detail Records

SNetAnalysis – Social Network Analysis Software as developed for thesis simulation

SQL – Structured Query Language

References

Research Papers:

[1] E. Bott , Urban Families: Conjugal Roles and Social Networks. 1995

[2] R.S. Burt, Models of network structure. In A. Inkeles, J. Coleman and N.Smelser, editors, *Annual Review of Sociology*,1980.

[3] A.S.Klovdahl, VIEW_NET. A new tool for network analysis. *Social Netwroks,*1986.

[4] J.L.Moreno, Who Shall Survive? Foundation of Sociometry, Group Psycology, and Sociodrama. *Washington D.C. Nervous and Mental Desease Monograph,* 1934.

[5] Xu Jennifer, Chen Hsinchun 2005. *ACM Transactions on Information systems*.

[6] Nasrullah Memon, Henrik Legind Larsen 2006. *Practical Approaches for*

Analysis, Visualization and Destabilizing Terrorist Networks.

[7] Christine Kiss, Andreas Scholz, Martin Bichler 2006. *Evaluating Centrality*

Measures in Large Call Graphs.

[8] Yonggu Wang and Xiaojuan Li 2007. *Social Network Analysis of Interaction in Online Learning Communities.*

[9] Lian Yan, Michael Fassino, Patrick Baldasare 2005. *Predicting Customer*

Behavior via Calling Links.

[10] Wei-Guang Teng, Ming-Chia Chou 2007. *Mining Communities of Acquainted Mobile Users on Call Detail Records.*

[11] Fu-ren Lin Chun-hung Chen 2004, *Developing and Evaluating the Social Network Analysis System for Virtual Teams in Cyber Communities.*

[12] Garton, L., Haythornthwaite, C., and Wellman, B. "Studying Online Social Networks," *Journal of Computer Mediated Communication* (3:1), 1997

[13] Wellman, B. "*For a social network analysis of computer networks: A sociological perspective on collaborative work and virtual community*," Proceedings of the 1996 ACM SIGCPR/SIGMIS Conference, Denver, Colorado US, 1996

Books and Articles:

[14] Scott Mitchell, from Trees to graph. January 2005.MSDN Library

[15] Eugene Lepekhin. Trees in SQL data bases. *www.codeprojects.com*

[16] Database Systems. *Principals, design and implementation.* Catherine Recardo

[17] Software Engineering. *A practitioner Approach.* Roger's Pressman

[18] Computer Graphics *with OpenGL*. Hearn Baker

[19] Stephen C. Perry. Core C# and .NET

[20] Sikha Saha Bagui, Richard Walsh Earp. Learning SQL on SQL Server 2005