MASTERS THESIS

A DATA MINING FRAMEWORK FOR SOCIAL GRAPH GENERATION AND ANALYSIS

by

Danish Kumar

Under the Supervision of
Dr. Muhammad Abulaish

Submitted in partial fulfillment of the requirements for the award of the degree of Master of Science in Computer Science

to the

DEPARTMENT OF COMPUTER SCIENCE
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
SOUTH ASIAN UNIVERSITY, NEW DELHI-110021, INDIA
May, 2018

Declaration

I hereby declare that the thesis entitled "**A Data Mining Framework for Social Graph Generation and Analysis**" being submitted to the Department of Computer Science, Faculty of Mathematics and Computer Science, South Asian University, Delhi in partial fulfilment of the requirements for the award of the degree of **Master of Science in Computer Science** contains the original work carried out by me under the supervision of **Dr. Muhammad Abulaish**. The research work reported in this thesis is original and has not been submitted either in part or full to any university or institution for the award of any degree or diploma.

Danish Kumar
SAU/CS(M)/2016/14

Acknowledgement

I turly indebted and thankful to Dr. Muhammad Abulaish,my supervisor, for his continous guidance and encouragement throughout the process of this thesis. It's a great privilege to complete this work under his guidance and supervision. His way of guiding always instilled in me enough confidence to think in depth on my research topic. His constructive comments and suggestions at each stage were immensely helpful in bringing this work to the present shape. His moral support and constant encouragement have been the main reason I was able to carry our this research.

I am also thankful to my friends and colleagues for their valuable insights and suggestions. Finally, it is my parents, brothers and sisters whose unconditional love and support give me the strength for carrying out the studies.

Danish Kumar

To my Uncle, Grandparents & loving parents ...

Abstract

Due to increasing popularity and easy accessibility of the social networking services, the number of users in social networks are increasing rapidly, and as a result, their size and user-generated contents are growing day-to-day. One of the requirements is to capture such huge amount of data and analyze them for desired purposes, such as target marketing, recommendar system design, open-source intelligence and cyber security. Twitter is one of the most popolar social network sites (aka microblogging site) and it is used by almost every person for news update, information sharing, viral marketing, etc using 280 characters.

In this thesis, I have proposed a text analytics framework to analyze Twitter data at different levels of granularity. One of the distinguishing features of the proposed framework is to exploit bot content and structural information for tweets analysis. The proposed framework first models the tweets into a multi-attributed graph, wherein tweets are represented as nodes and inter-tweet relationships are represented as edges. For node labeling, we have used NLP techniques to identify features from tweets, whereas edges are labeled using meta-data (such as hashtags, mentions, followers, etc.) that are common to the tweets connected by the edges.

For analyzing multi-attributed graph, I have considered two algorithms, MAG-Dist and MAG-Sim, to convert multi-attributed graph into a simple and similarity graph. Finally, I have applied MCL (Markov Clustering) algorithm to cluster the nodes of the multi-attributed graph, each group consisting of a particular subset of tweets representing an event. The experimental evaluation of the proposed approach is done on a real dataset crawled from Twitter.

CONTENETS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ICT	Information and Communication Technology
NLTK	Natural Language Tool Kit
MAGDIST	Multi-Attributed Graph Distance
MAGSIM	Multi-Attributed Graph Similarity
MCL	Markov Clustering
TF	Term Frequency
IDF	Inverse Document Frequency
WAG	Weighted Attributed Graph
iPaM	Index Based Pattern Matching
TPR	True Positive Result
FPR	False Positive Result

Chapter 1

Introduction

1.1 Overview

Here in this chapter, I explain a brief introduction about data mining framework for social graph generation and analysis. Data Mining is the techniques of information extraction from a huge collection of data, there are various application for the extracted information as follow:

- Fraud detection
- Market analysis
- Prodution control
- Science exploration (Bio informatic)
- Target marketing

Due to advancement in the Information and Communication Technology (ICT), especially the invention of smartphones and easy accessibility of it to the people around the world. World become a globle village and people are connected via different social media platforms called social networks. There various types of social network such as Facebook, Twitter, Google+ etc. We consider our work of analysis for Twitter social network. Twitter allows user to tweet (a text message with max lenght of 140 charachters) and tweets can contain text of length 140 characters,images and videos.All users of twitter can see the post without any constraint. In twitter one can follow others without following confirmation this made twitter a fast social media for news viral. It has more than 336 million users (as per 2018) actively. Smartphones and other web apps are user-friendly, even a person can use the social networking services with bit knowledge of operating a smart-phone. This lead to rapid growth of data in social networks. Now the challenging task is how to use these data usefully and extract useful information. For this purpose, we worked on data mining framework for social graph generation and analysis where our aim is to find clusters in social networks of different objects, so further that can be used for more analysis purpose.

1.2 Problem Definition

The data on the internet is unstructured data, the way to organize the data in structured form we use graph. Rely on the nature of data to be modeled, the graph could be directed/undirected or weighted/un-weighted. As like there are many complex data such as online social network, in which an entity is represented by set of features and multiple relations exists between an entity pair.

The problem of my thesis is about to genrate social graph as multi-attributed graph, and by analysis to decompose a generated graph into multiple cohesive sub-graphs, called cluster, based on some common properties. The social graph is generated from the Twitter (social network), on the basis of crawling some countable tweets from different four events. The tweets are preprocessed by NLTK, in which stemming, punctuation, stop words removal, and URL removal processed. With only consideration of text, hashtags and timing of tweets, all the other unnecessary things removed by NLTK.

After NLTK, the top most words extracted (in descending order) from tweets with the help of tf-idf proces. Top most words are represented as k dimensional binary vector for each vertex, and overlapping of number of hashtags and time, represent the multi-edges with values, for generate multi-attributed social graph. For analysing the graph, I used clustering, because the field of clustering multi-attributed graph is still unexplored. To clustering the multi-attributed graph, I used two proposed algorithms (MAG-Dist and MAG-Sim) to convert multi-attributed graph into simple and similarity graph because similarity measure is the key requirement for any clustering algorithm.

I used Markov Clustering (MCL) for clustering the multi-attributed graph on Iris and twitter data set.

1.3 Scope of Proposed Work

Social media analysis has been the most trending research topic and analysing the real world issue with the help of virtual world is not an easy task. The scope of proposed work is to how to generate multi-attributed social graph from the social network. Multi-attributed graph is used to model many complex problems, mainly those in which entities are characterized using a set of features and linked together in different

ways. For example an online social network in which user can be characterized using set of features like ID, display name, interest etc. For the analysis, clustering technique is introduced, the main goal of the clusetring is to partitioning the large graph into several densly connected components. I used markov clustering algorithm for clustering the data, it is fast, scalable and efficient algorithm for clustering. Application for graph clustering include community detection in online social network, spammer identification and in protein interaction network etc.

The current need for the clustering of the multi attributed graph is in many various domains and as most real-world problems and conceptual information can be modeled as multi attributed graphs. Where objects of the problem are to be considered as vertices of the multi attributed graph and the interaction between the nodes or data objects can be represented as the set of edges of the multi attributed graph.

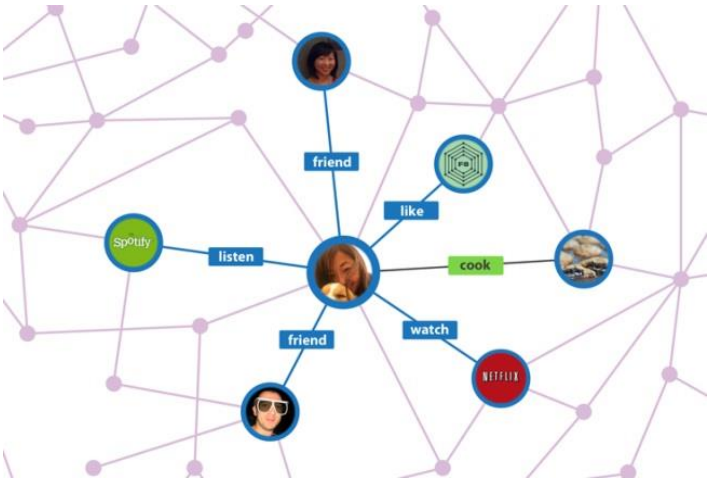

Figure 1.1 An exemplar social graph

1.4 Thesis Outline

The rest of the chapters are structured as follow:

Chapter 2 the preliminaries here it defines the concepts that are used in thesis and in my work, which will give a brief to the reader to understand and read the rest of chapters comfortably.

Chapter 3 is the literature review, this chapter gives an introduction to the related works done on the same topic

Chapter 4 is proposed work, this chapter explains the methods that are used in experiments for analysis and generation of the social graph.

Chapter 5 is the experimental result and evaluation, this chapter explains step by step the experiments over two datasets.

Chapter 2

Preliminaries

2.1 Introduction

In this chapter, I am introducing some terms that I am using in the rest of thesis. It helps for understanding those terms which are used in further chapters.

Definition 2.1. A *graph* 'G' is a set of vertex 'V' called Nodes that are connected by edges 'E' called Links. Mathematically we can say G = (V, E). Graph is a way to formally represent a network or collection of interconnected objects. A graph is a powerful tool for modeling database objects and their relationships among data items in various application domains.

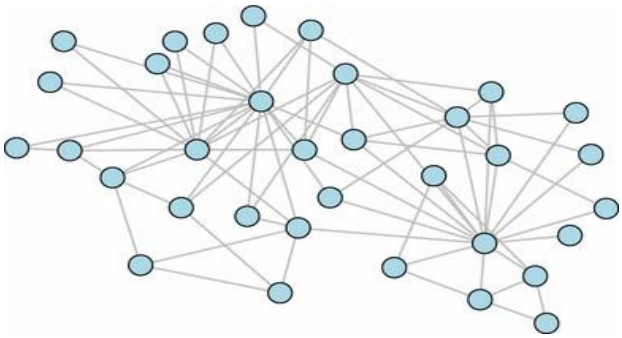

Figure 2.1 An examplar graph

Definition 2.2. An *Edge-Weighted graph* always having weight on their edges. Mathematically we can say G= (V, W), where G= (V, E) is a graph where W: E→R is a weight function.

Definition 2.3. A *Node Attribute Graph* always having an attribute on the particular node. Attribute defines the properties of nodes.

Definition 2.4. A *Multi Attributed Graph* is defined as, a graph having some attributes on the nodes. Attributes are the properties of nodes.

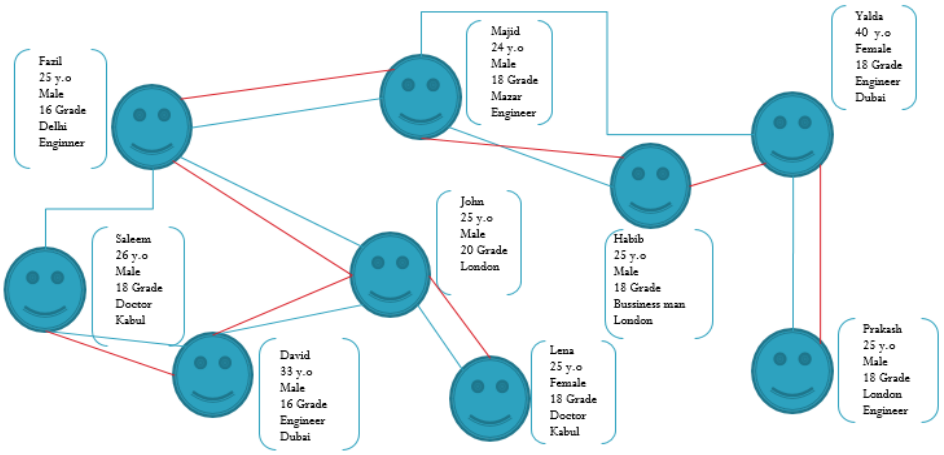

Figure 2.2 An examplar multi-attributed graph

Definition 2.5. A *Social Network* is defined as a chain of personal and individual connection. It is based on the interconnection of people to each other.

Example: It establishes online communities that help people. E-mail system where people can send and receive their data. Facebook, Twitter etc. are the social networks.

Definition 2.6. A *Pattern Matching* is a technique to find sub-graph of a given data graph G which match our query graph. It works by identifying match and ranking match.

Definition 2.7. A *Clustering* is a process of organizing objects in a group by similarity. Clustering can be done by distance, value etc.

Definition 2.8. *Markov Clustering* is a clustering algorithm based on the simulating a random walk on the weighted graph. The intuition of this MCL is that if node transition reflects the weight on edges, then transition from one node to another node within a cluster are much more likely than the transition between nodes from the different cluster.

Definition 2.9. A *Norm* is a function that assigns a strictly positive length or size to each vector in a vector space.

Definition 2.10. A *Frobenius Norm* sometimes also called as *Euclidean Norm*. It is a matrix of m×n defined as the square root of the difference of Absolute Square of elements.

$$\|M_t - M_{t-1}\|F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}(M_t(i,j) - M_{t-1}(i,j))}^{2}$$

Definition 2.11. The *tf (Term Frequency)* measure that how a term occurs in a document. We calculated as.

$$\text{tf} = \frac{Number\ of\ term\ t\ \ appears\ in\ a\ document}{Total\ number\ of\ terms\ in\ the\ document}$$

Definition 2.12. The *idf (Inverse Document Frequency)* measures how important a term is. We can calculate it as

$$\text{idf} = \log \frac{Total\ number\ of\ \ documents}{Number\ of\ documents\ with\ term\ t\ in\ it}$$

Definition 2.13. A *Similarity Matrix* is a matrix of scores that represents the similarity between the pairs of data points.

Chapter 3

Literature Review

This chapter is related to the extraction of knowledge through different proposed work.

Whenever we think about social networks, it always comes to our mind about Facebook, Twitter, Hangout or any other communication website all these are the platform of the social network. The social network is actually a network of social interactions, a chain of individuals and their personal connections. In the social network, there is a network in which differently collected entities (usually peoples but they could be something else entirely) that participates and there should be at least one relationship between entities such as in Facebook we call that relation as a friend. Naturally, the Social network is modeled as a graph which we refer to "Social Graph" sometimes. Entities are the nodes and an edge represents the relationship between two nodes. Often social networks are undirected (Facebook friends) but they can direct like as followers on Google+. We have different types of Social network such as:

- Telephone Network
- Email Network
- Collaboration Network

The author in [1] talks about social graph generation, it can be modeled as Static Graph Modeling and Dynamic Graph Modeling. Social networks are implanted in many types of data at many different scales on the basis of the text, communication system, sensor networks and social media etc. Author constructed a social graph by text, accomplished in different ways. First, the link approach used upon the co-occurrence of entities then look for mentions of relations in text and finally co-occurrence resolution.

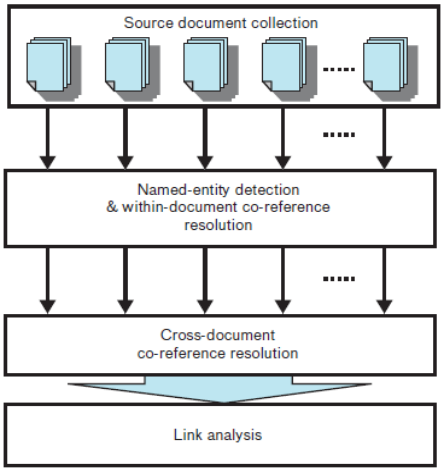

Figure 3.1 Process of information extraction from a text corpus for social network construction

After retrieving information from original data it must be stored in a structured form so that analysis will be easy. Multiple possible structures for the representation are mathematically equivalent but the main difference is that they arise in multiple fields. So attributed graph is an alternative solution for them. The author talks about the clustering of Social Network graphs, as we know there is an important factor of a social network is they contain communities of entities that are connected by many edges and clustering in the social graph is an active research area.

For retrieving the top-k results in a graph there is pattern matching technique. It is very difficult to find the optimal results from graph because graph has many properties and structure. So come up with all the issues, the work done in [2] focuses on the pattern matching on the weighted attributed graph (WAG) based on the Pattern queries, Ranking, Indexing and Matching of the indexed-based pattern matching(iPaM). There are some problems with pattern matching on a WAG such as structure, algorithms (exact vs. inexact), queries (Point vs. Range) and solutions (Optimal vs. Approximate). To address all that issues the technique introduced named as iPaM. It takes as input a graph G= (V, E) and a matrix W. The Matrix W contains entries Wij corresponds to the weight of attribute j on node i. Pattern query applied on a graph is also a sub-graph, and defined by Hq = (V', E') and Wq (a node x attributed matrix, containing weights for each attribute). Hq is a small graph extracted from types of queries like Point Query (contains discrete values between [0, 1] and the sum of a row of Wq must be up to 1), and Range Query (flexible querying, the user doesn't need to specify the weight of each attribute or node).

After giving a query the task is to return top-k best matches, for this process iPaM needs to rank the result according to divergence on graph structure and on weighted Attributes. Jensen difference is the divergence function that is used in ranking. iPaM builds and maintains an index structure offline, which is used to speed up pattern matching during query time and provides the matching over weighted attribute and structure of the WAG. iPaM matching will match the results with the given query.

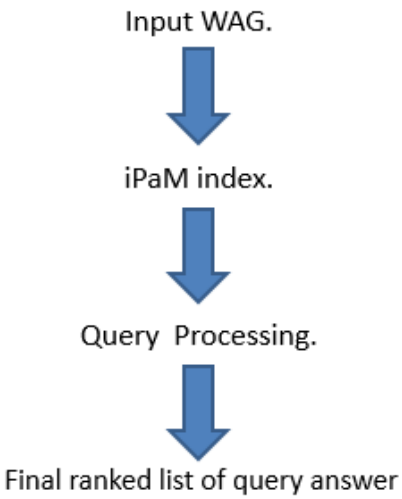

Figure 3.2 Work-flow of iPaM

The author investigates different problems with iPaM regarding structure, large graphs. In [3] authors propose a pattern matching problem over a large directed graph (based on reachability constraints) and corresponding approximate matching algorithm to select the best set of patterns when an exact match is not possible. iPaM comes with all the solutions to finding an optimal match for given pattern query and this process is NP-complete. Future work includes the extension of iPaM for time-evolving graphs.

Applications of iPaM are vast in the fields of Social Media, Google Map (find the shortest path), Citation Network, Ranked based Analysis. In Co-authorship network, with the help of iPaM, it is very easy to find the authors with a specialty of the research area from the network. It will be more beneficial for the graph indexing.

For the analyzing social network, Data mining techniques are applied. We are familiar with the communities of networks and for analyzing communities detection, clustering is the most used technique all around research area. In [4] author describes

the clustering of the graph and its applications, basically, clustering is to divide the graph into subgraphs on the similarity basis. Clustering can be done with respect to many aspects such as centroid based, connectivity based and density based etc. There are so many applications of clustering, such as:

Image Processing: in which detecting distinct kinds of pattern in image data.

Data Transformation: conversion of vector data into graph format and vice versa.

Biological Networks: classification of gene expression data and protein interaction.

The author also discussed there are some problems regarding the graph clustering, which are Parameter Selection (How is the user to determine parameter values for input in clustering algorithm), Scalability (Run Time and Memory consumption) and Evaluation (How we decide which of several clustering is best). For the future work, clustering can be applied to a weighted graph, directed graph, Multi-graphs, and Hypergraphs.

In [5] 'Automatic Subspace Clustering of High Dimensional Data for Data Mining' authors describe the types of clustering. Clustering in detailed to describe the homogenous group of objects according to their attributes or dimensions. Nowadays clustering is classified on two techniques namely, Partitional and Hierarchical.

Partitional Clustering: It is simply dividing the set of data objects into a non-overlapping subset (that subset is a particular cluster), so each object is in exactly one subset.

Hierarchical Clustering: It is set of the nested cluster. In this clustering, it doesn't assume a particular value of k, as it is needed in k-means clustering. It also follows two ways, Agglomerative (Bottom-Up) and Divisive (Top-Down).

Authors works on the subspace clustering, it is an extension of normal clustering that seeks to find clusters in different subspaces within a cluster. There are so many applications for clustering in the Information Integration System, Text Mining and Bioinformatics. Clustering algorithms have been used in DNA Microarray data for identification and characterization of disease.

In [6] authors describe the weighted distance measure for Multi-Attributed graph, by two proposed algorithm i-e MAG-Dist and MAG-Sim algorithm. MAGDist (Multi-

Attributed Graph Distance) it calculates the distance between each vertex pairs on the Euclidean Norm basis, after calculating the distance the other algorithm MAG-Sim (Multi-Attributed Graph Similarity) that generates a similarity graph on the obtained result which can be analyzed using classification and clustering algorithm MAG-Dist calculates distance with respect to the attributes values and it also considers the edges weights of the graph.

In [7] Data Mining and Analysis Fundamental concepts and Algorithms, authors describe clustering and different techniques of clustering (i-e K-means Clustering, Markov Clustering etc.). Authors also explain that graph clustering is the special case of clustering which divides an input graph into a number of connected components (sub-graph), such that intra-component edges are maximum and inter-components edges are minimum. Every connected component is a Cluster. Nowadays graph clustering got the attention of many researchers, and multi-attributed graph clustering is still unexplored. For the experiment in [6], they tested algorithms over the Iris data set and Twitter dataset. Where Iris dataset contains total 150 instances of different three categories of Iris flower, and the data represented as 150 × 5 data matrix. They also analyzed the significance of different similarity graph generation in terms of True Positive Rate (TPR) and False Positive Rates (FPR). For the Twitter dataset, they used tf (Term Frequency) and idf (Inverse Document Frequency) analyzing the top words from the data set. For the attributes, they used those words and compare them with every node, finally create a binary vector for every node as a weighted attributed graph. Future work refers that by applying the proposed algorithm on social network and citation network to analyze them at different levels of granularity.

In [8] 'Numerical Analysis Lecture Note', the author explains the Numerical Analysis part such as Matrix Algebra, Eigen Values, and Numerical Solutions. By this paper author from [4] derived equations for MAGDist and MAG-Sim algorithms with the help of vector properties.

Later by the visualization of the graph, we used a new platform named as Neo4j. [9] Neo4j is a world's leading graph database and most active graph community. In Neo4j user can store trillions of entities of the largest dataset, and the data is stored in the form of node, edge, and attribute, each node, and edge can have any number of the attribute. For retrieving the result, the declarative graph query language named "Cypher" is designed to visually represent relation and graph patterns of nodes. Like

SQL, Cypher query allows the user to what actions they want to perform such as Match, Insert, Update or Delete on their graph data. This is an open graph query language because Ne4jo Inc. wants to make it most popular graph query language with the aim that cypher becomes the SQL for graphs. Neo4j gives users the ability to inspect the animated graph by zooming and panning across visible domain and also edit properties, nodes, and relationship. By selecting and toggle particular node user can understand the properties, attributes and adjacent nodes by relationship.

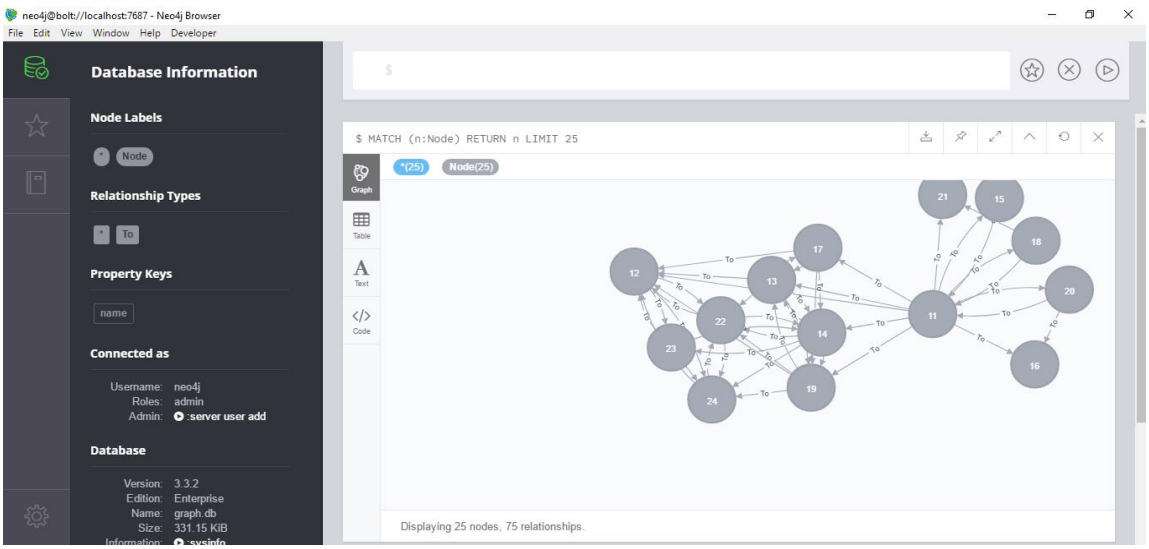

Figure 3.3 Visualization of a graph in Neo4j

[10] The Google Page Rank is described to find out the webpage's popularity score. The Page Rank formula presented in front of the world in Brisbane at seventh World Wide Web conference (WWW98) by Sergey Brin and Larry Page (founders of Google in 1998). There is an algorithm for calculating the score of the webpage and it is an iterative process. Page Rank also used in the graph field to calculate the popular nodes (having more number of in-degrees). Webster starts a random webpage, whenever he visits a web page, the randomly hyperlink on that page chosen by him. The Web pages with hyperlinks between them are viewed as directed graph called hyperlink graph. If there are some web pages P_i and P_j and there is a hyperlink that points from P_i to P_j called Outlink and hyperlink that points from P_j to P_i called Inlink. For the representation, a matrix will be created, called Hyperlink matrix (aka Adjacency Matrix) of the graph. For the popularity score, a link from more valuable page to user's page is more important and link from a page having more outlinks to

user's page is less valuable. To calculate the popularity score there are some steps followed by the algorithm:

1. The Hyperlink Matrix.
2. The Stochastic Matrix.
3. The Google Matrix.
4. First Iteration.
5. More Iteration.

Chapter 4

A Data Mining Framework for Social Graph Generation and Analysis

4.1 Introduction

Data Mining is the process of discovering insightful, interesting, and novel patterns, as well as a descriptive, understandable, and predictive model from large-scale data. Data mining is the part of larger Knowledge Discovery Process in which pre-processing tasks (such as data extraction, data cleaning, data reduction and so on) and post-processing tasks (such as pattern and model interpretation, hypothesis confirmation and generation and so on). Data is growing on internet day by day, it is very hard to extract the data from the large database, and there are some techniques to analyze the data. Data mining involves six common classes of tasks.

- **Anomaly Detection:** Identification of unusual data records.
- **Association Rule Learning:** Searches relationship between variables.
- **Clustering:** Discovering groups of similar data.
- **Classification:** Task to generalize known structure to apply to new data.
- **Regression:** Attempt to find a function which models the data with least error.
- **Summarization:** Providing a more compact representation of the data set, including visualization and report generation.

By the growing exponentially of data it is not easy to represent the data by space and memory saving, so by the graph it is easy to represent. Social networks are naturally modeled as a graph which we sometimes refer as a Social Graph. Nodes are represented by entities and edges denote the relationship between them. Social graphs are generated by different ways on their aspects, like as distance, values, etc. Social media analysis has very vast research area; our main goal was to clustering for social network (Multi-Attributed) data set. Nodes having attributes along with weights and multi-edges with weights are determined by the algorithms (MAG-Dist and MAG-Sim) [5], Firstly by MAG-Dist calculates the distance between each vertex pair with the values of attributes and values of multi-edges. After calculating the final result of MAGDist algorithm, MAG-Sim generates Similarity matrix of that result. As we

know that similarity is the basic need of clustering so we used these algorithms for calculating similarity matrix of the multi-attributed graph.

4.2 MAG-Dist

It reads the multi-attributed graph a list of vertex and edge vectors as two separate CSV files and calculates the distance between each vertex pair.

$$\lambda = \frac{1}{(1+\omega(u,v))\gamma}$$

This equation calculates the distance between a vertex pair with ωthat represents the aggregate weight of the edges between vertex pairs.

$$\Delta(u,v) = \sqrt{\lambda} \times \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2}$$

This equation is based on Euclidean norm and it calculates the distance values. Where λ is a scalar value that depends on the aggregate weight ω**.**T his equation is monotonically decreasing function.

4.3 MAG-Sim

It generates similarity graph using the distance values calculated by the MAGDist algorithm. It reads each vertex pair and its distance value and then calculates the similarity between each vertex pair.

$$Sim(i,j) = 1 - \frac{\Delta(i,j)}{max_{x,y \in v}\{\Delta(x,y)\}}$$

After applying these algorithms the result will be in similarity graph and that can be analyzed using classification and clustering algorithms. Analyzing of social media can be done in different techniques such as clustering, anomaly detection, and summarization etc. There are so many techniques of clustering but we are doing Markov Clustering. The main intuition of using Markov clustering is because of a random walk on the graph by alternating two operators: Inflation and Expansion.

4.3 Markov Clustering

Markov clustering is an unsupervised algorithm. It simulates on the basis of a random walk on the graph. The basic intuition of Markov clustering is that if node transitions reflect the weights on edges, then transition from one node to another node within a cluster is much more likely than the transition between nodes from the different cluster because node within a cluster has higher similarities or weights and nodes across cluster have lower similarity.

It works by the two operations: Inflation and Expansion. When the node is assigned a weight after the complete walk, a graph will be generated and that graph will be translated into the stochastic matrix. The stochastic matrix represents the transition probabilities between all pairs of nodes. The probability of random walk of length n between any two nodes can be calculated by increasing the value of matrix to the exponent n, and this process is called expansion. Markov clustering further overstates this effect (expanded matrix) by taking exponent of every entity of matrix and then rescaling each column so that it remains stochastic this process is called inflation. By simulating both processes alternatively, clusters are identified.

$$\gamma(M, r) = \left\{ \frac{(m_{ij})^{\gamma}}{\sum_{a=1}^{n} (m_{ia})^{\gamma}} \right\}$$

This equation is transition probability inflation which states that when we consider a variation of the random walk where the probability of transitioning from node i to j is inflated by taking each element to the power $\gamma \geq 1$.

4.3.1 MCL Algorithm

Markov clustering algorithm is an iterative method which used for clustering and it works on the expansion and inflation of the matrix. Matrix expansion parallel to taking the successive power of transition matrix and other side inflation of the matrix creates higher probability transition even more likely and reduce lower probability transitions. Clusters depend on the value of inflation ($\gamma \geq 1$), higher values gives more, smaller clusters whereas smaller values gives fewer and larger clusters. However, it can't be pre-determined that how the exact number of clusters will be.

MCL firstly adds self-edges (if they don't exist), then it generates Markov matrix ($M_t=\Delta^{-1}A$, Δ is a degree matrix). After creation of Markov matrix, it starts the iterative process with inflation and expansion until transition matrix convergence (the difference between transition matrix and two successive iteration falls under some threshold $\epsilon \geq 0$). The stopping condition of MCL is Frobenius norm.

4.4 Analyzing Social Graph

For analyzing social media, the worked on Twitter (social Platform) and for analyzing, clustering applied on that social network. As there are so many techniques for clustering but we applied Markov Clustering because it has more efficiency than others.

4.4.1 Crawling Data from Twitter

There are two ways through which twitter data can be retrieved: (a) Standard search API, (b) Enterprise search API. Standard search API is free to use. Standard search API returns a collection of relevant tweets matching a specific query. It returns tweets posted within the last seven days. We will be using standard search API for collection of tweets about different events.

4.4.2 Pre-Processing of Tweets

The tweets need to be filtered because of presence of unnecessary things such as Stop words, Punctuation and URL. Hashtag needs to be separated because it can be used to form the relationship between the nodes.

4.4.3 Creating Multi-Attributed Graph

After second step the graph of tweets will be created on the basis of hashtag overlap and time overlap. It will be a simple graph with multi-edges. To convert simple graph into a multi-attributed graph, the topmost words will be taken as attributes (in the form of binary k dimensional vector) on the basis of binary values. The topmost words will be extracted with the help of tf-idf.

4.4.4 Convert Multi-Attributed Graph into Similarity graph

Two proposed algorithms [5] MAG-Dist and MAG-Sim will generate similarity graph. MAGDist algorithm will find the distance between every pair of nodes along

with weights of multi-edges. MAGSim will use the result of MAGDist and will generate Similarity Graph. These two algorithms will be used to convert multi-attributed graph into similarity graph.

4.4.5 Clustering

After similarity graph it is easy for clustering, Markov Clustering (MCL) technique can be applied to the similarity graph to make a cluster for every event.

Chapter 5

Experimental Setup and Results

5.1 Introduction

To experience, that how we get the clustering result from above-proposed methods. And check the efficacy of the proposed methods. I apply methodology for the clustering of real-world datasets. I use labeled dataset, so I can use label to match it with cluster result data to find the accuracy of the proposed methods.

5.2 Iris Dataset Clustering

Iris data set is famous multivariate dataset which was created by R.A Fisher as an example for discriminant analysis. The data explain four different characteristics of 3 iris flower spices, sepal length, sepal width, petal length and petal width. Where iris dataset contain such 50 observation of each spice and total instance of iris dataset is 150 for every four characteristics. The dimension of iris dataset is 15 x 5, as 4 columns keep characteristics measures and 1 column keeps the spice labels as given in below table:

Table 5-1 Iris Data Set

Iris-setosa	5.1	3.5	1.4	0.2
Iris-setosa	4.9	3	1.4	0.2
Iris-setosa	4.7	3.2	1.3	0.2
Iris-setosa	4.6	3.1	1.5	0.2
Iris-setosa	5	3.6	1.4	0.2
Iris-versicolor	7	3.2	4.7	1.4
Iris-versicolor	6.4	3.2	4.5	1.5
Iris-versicolor	6.9	3.1	4.9	1.5
Iris-versicolor	5.5	2.3	4	1.3
Iris-versicolor	6.5	2.8	4.6	1.5
Iris-virginica	6.3	3.3	6	2.5
Iris-virginica	5.8	2.7	5.1	1.9
Iris-virginica	7.1	3	5.9	2.1
Iris-virginica	6.3	2.9	5.6	1.8
Iris-virginica	6.5	3	5.8	2.2

I want to check the efficiency of proposed method for clustering the Iris dataset. To do this, we need to go through following steps in sequence:

- Consider each instance of dataset as vertex
- For edge, we use Gaussian Similarity as described bellow

Generate edges between every two instant pairs if their Gaussian similarity greater than 0.55

4.2.1 Gaussian Similarity

$$e(u,v)=\begin{cases} k^G(u,v)\ if\ k^G(u,v) \geq 0.55 \\ 0\ otherwise \end{cases}$$

$$k^G(u,v) = e^{\frac{-\|u-v\|^2}{2\sigma^2}}$$

Algorithm: Gaussian similarity $(irisdata\ matrix)$

For u in the dataset

  For v in the dataset

    Compute $k^G(u,v)$

    If $k^G(u,v) \geq 0.55$

      e (u, v) = $k^G(u,v)$

    else $e(u,v) = 0$

  end

end

I do not compute Gaussian similarity for diagonal elements; I directly add 1.0 for diagonal as the similarity of an element with itself is 100 %. The resulted graph will look like as shown in figure 5.1.

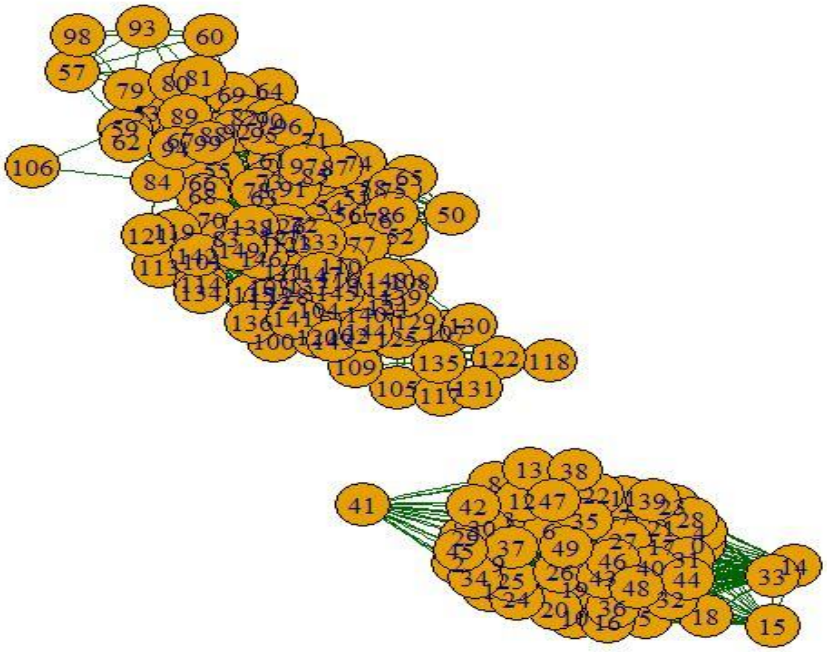

Figure 5.1 Generated graph using Gaussian similarity

Now, we modelled the iris dataset as graph as shown in Figure 5.1. Here after we want to use MCL (Markov Cluster) algorithm to cluster the iris dataset. Here for this purpose we consider each instance as vertex as mention above and edges between the vertices pair. And also here we only consider the structural part of the modeled graph and we do not consider the vertices attribute value in clustering process. We do this to evaluate the MCL result for Gaussian Similarity modelled graph and later for MAG-Dist (Multi-attributed graph distance measure) and MAG-Sim (Multi-attributed graph similarity) algorithms modelled graph. We do this to know the performance of the method we chosen for our task. So now, we feed the result of Gaussian Similarity in matrix form to MCL algorithm. The MCL produce the result as follow in figure 5.2.

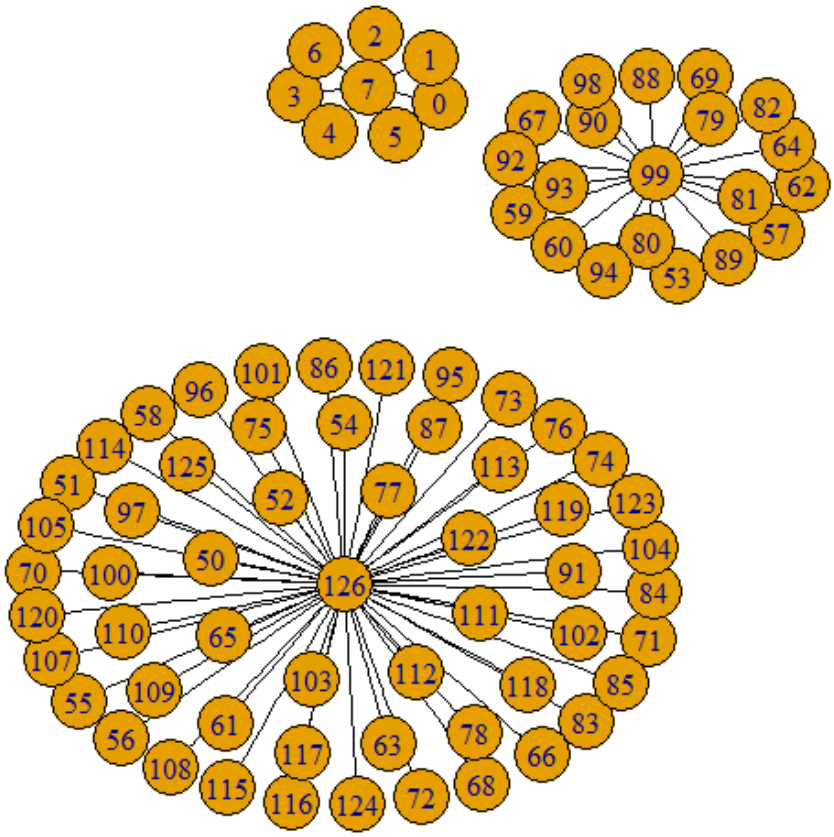

Figure 5.2 MCL Clustering result for Gaussian kernel result

Table 5-2 Entities in cluster

	Cluster 1	Cluster 2	Cluster 3	Total
Setosa	8	0	0	8
Versicolor	0	20	30	50
Virginica	0	0	27	27

After I use the proposed method MAG-Dist (Multi-attributed graph distance measure) and MAG-Sim (Multi-attributed graph similarity) measure. As described in chapter 4, this method consider both part of multi-attribute graph. First with consideration of

both attributes and edges weight, it calculate the distance between two vertices pair. And then MAG-Sim algorithm calculate the similarity using the difference measure output from MAG-Dist algorithm. As follow:

Calculate the similarity for the attributed graph after using the normalized result of distance measurement.

$$Similarity = 1 - distance$$

And we add edges between each vertex pair if the similarity was greater than 0.85. We now we get a new simple graph. As shown in the following figure 5.3:

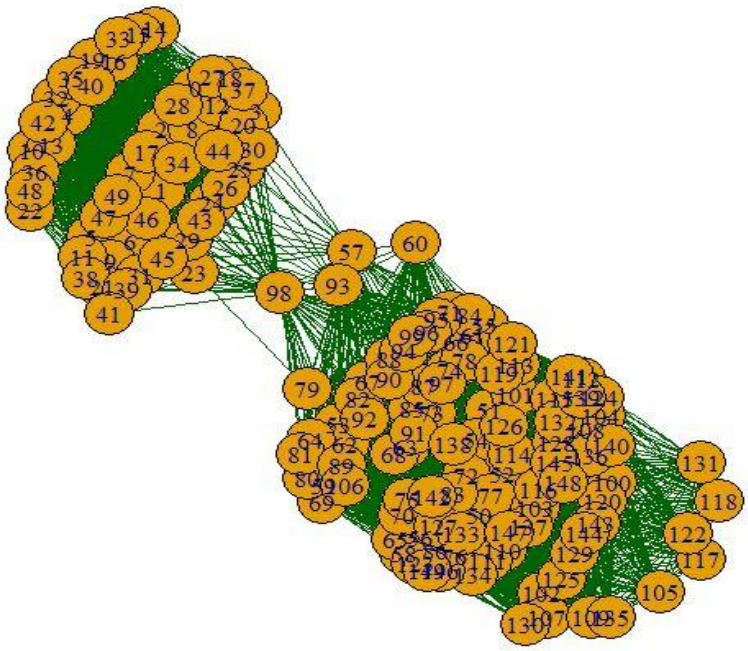

Figure 5.3 Similarity graph after proposed methods

Again here I use MCL (Markov clustering) algorithm to cluster the result new simple graph. And with applying MCL after 8 iteration we get the following result as shown in figure 5.4 for iris data set.

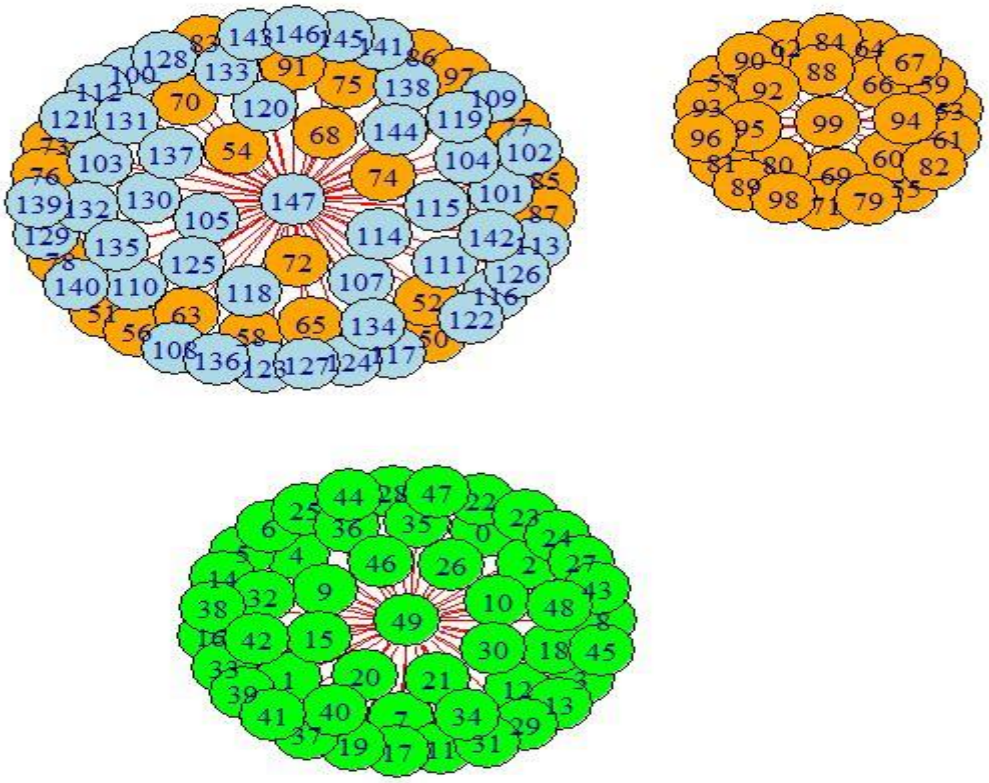

Figure 5.4 MCL clustering result after distance measure similarity graph

Table 5-3 Cluster of iris data set

Cluster 1		Cluster 2		Cluster 3		Total
setosa	50	Setosa	0	Setosa	0	50
Versicolor	0	Versicolor	27	Versicolor	23	50
virginica	0	Virginica	0	Virginica	48	48
Total used Nodes						148

5.3 Tweets Clustering Based on Events

In order to evaluate the performance of the proposed methods for the clustering of the attributed graph. And to practically see the performance of the proposed method for

the clustering the tweets of twitter social network .We did experiment on tweet real dataset. We have collected 50 tweets, where these tweets relate to the four different events, as detial given follow in table 5.4.

Table 5-4 Tweets related to various events

Events	No.of tweets
#SpringFest	10
#Afghanistan news	10
#PashtunTahafuzLongMarch	20
#RamadanMonth	10
Total Tweets	50

As it is so common for social network users to type the sentence in not proper structure and they do not give importance to the grammar to write the proper sentence. the most important to them is to spread their views rather than writing Grammarly correct sentence and correct typing, and also social network users are now adopted in reading and understanding of such unstructured sentence and Grammarly uncorrected sentence in social networks. So because of this, all the time we get tweets from the twitters app, it will contain punctuations, stop words, hashtags, abbreviations and slang language. This creates a bottleneck for the processing of such data. So, for this reason, we consider the natural language preprocessing for our tweet dataset to remove those unnecessary part of tweets which either can affect our result and degrade the performance or does not help the performance improvement and also add more to improve the feature extraction. After this phase, we got preprocessed tweet dataset. hereafter we called it corpus which contains preprocessed tweets without having an unnecessary part as like original tweets, and then we did different feature extraction from the corpus which after we use for the generation of the social graph which has the form of an attributed graph. after these processes, we use to visualize the result of each step, as almost all our results are in the form of a graph. Hence we use networks python library to plot the graph for our resulted edges pair. Below, it explained in details for each step:

5.3.1: Data Collection

The data was gathered from the Twitter social network, Twitter is micro blogging service, which permits its user to post tweets, a status message which can have maximum 280 characters which usually use to carry personal views, news information, events information and information related to the different topics.

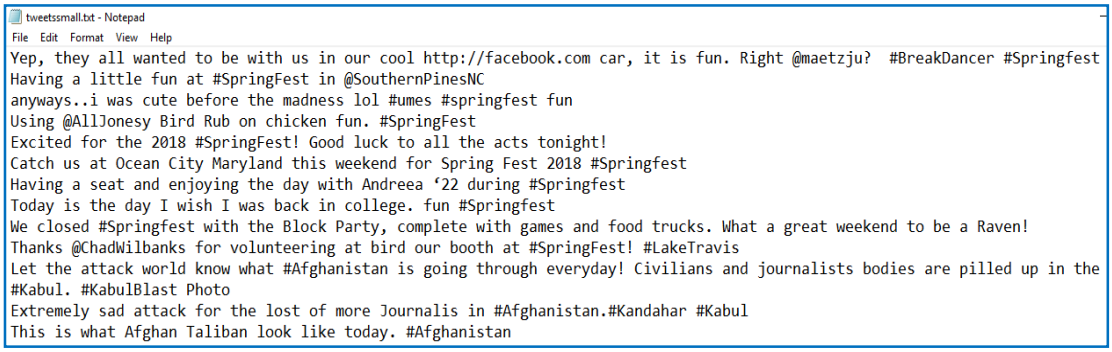

Figure 5.5 Sample tweets

5.3.2 Natural Language Pre-processing

Text can come in various forms as like a list of individual words, to sentences to multiple paragraphs with special characters (like tweets for example). so the text is semi-structured or unstructured data which do not reside in fixed field or record, thus we need to transforming text into some structure that an algorithm can digest it a complicated process. There are four different parts which help us to transform text into a form where the algorithm can deal with it:

- Cleaning consists of getting rid of the less useful parts of the text through removing of stop words, consideration of texts with capitalization and characters and other details while using it in the program.
- Annotation includes the application of a scheme and structure to texts. Annotations may include structural markup and part-of-speech tagging.
- Normalization consists of the translation (mapping) of terms in the scheme or linguistic reductions through Stemming, Lemmatization and other forms of standardization.
- The analysis consists of statistically probing, manipulating and generalizing from the dataset for feature analysis.

Here for tweets dataset, we did text data cleaning process which is consist of the following process:

1. URLs removal
2. Punctuation removal
3. Stopword removal
4. Stemming

5.3.3 URLs Removal

In order to remove the URLs (unified resource locators) or web address. Where it does not carry any information to help our clustering process improvement. So in closing first we remove the available URLs in tweet dataset.

```
>>> text="Yep, they all wanted to be with us in our cool http://facebook.com car, it is fun. Right @maetzju?  #BreakDancer
>>> texitWithOutLink=removelink(text)
>>> texitWithOutLink
'Yep, they all wanted to be with us in our cool  car, it is fun. Right @maetzju?  #BreakDancer #Springfest'
>>>
```

Figure 5.6 Result after URLs removal

5.3.4 Punctuation Removal

As like all language writing use punctuation, in English language writing we also use punctuation which help the readers to understand the text as like question mark(?) use for interrogative sentences ,exclamation mark(!),dots(.) use at the end of paragraph or sentence and comma which use to separate various part in a single sentence. Her feature extraction from tweets we do not consider this punctuation. So used to remove the punctuations.

5.3.5 Stop Words Removal

A majority of the words in a given text are connecting parts of a sentence rather than showing main subjects, objects or the intent. Words like “the”, “and” or other English stopping words which have the connecting job in a sentence can be removed by comparing the text to a list of stop word.

we used Python programming language for our implementation which has NLTK (Natural Language Toolkit) library for text data pre-processing and NLTK library by default has the list of stop words for the English language, where we used the list of default stop words and we also added some stop words into list which were not

available after we noted in our tweet dataset.

```
>>> text="Yep, they all wanted to be with us in our cool http://facebook.com car, it is fun. Right @maetzju?
>>> texitWithOutLink=removelink(text)
>>> texitWithOutLink
'Yep, they all wanted to be with us in our cool  car, it is fun. Right @maetzju?  #BreakDancer #Springfest'
>>> DelPuncAndStopwords=text_procss(texitWithOutLink)
>>> DelPuncAndStopwords
'Yep wanted cool car fun Right maetzju BreakDancer Springfest'
>>>
```

Figure 5.7 Result after stop-words and punctuation removal

5.3.6 Stemming

This method is used to identify the root/stem of a word. For example, the words connect, connected, connecting, connections all can be stemmed from the word "connect". The purpose of this method is to remove various suffixes, to reduce the number of words, to have accurately matching stems, to save time, memory space and also it helps to have a single root for different variants of a term. stemming can improve the features extraction, specifically the of-idf a feature that we consider in our implementation. We use the ported stemmer algorithm in our implementation.

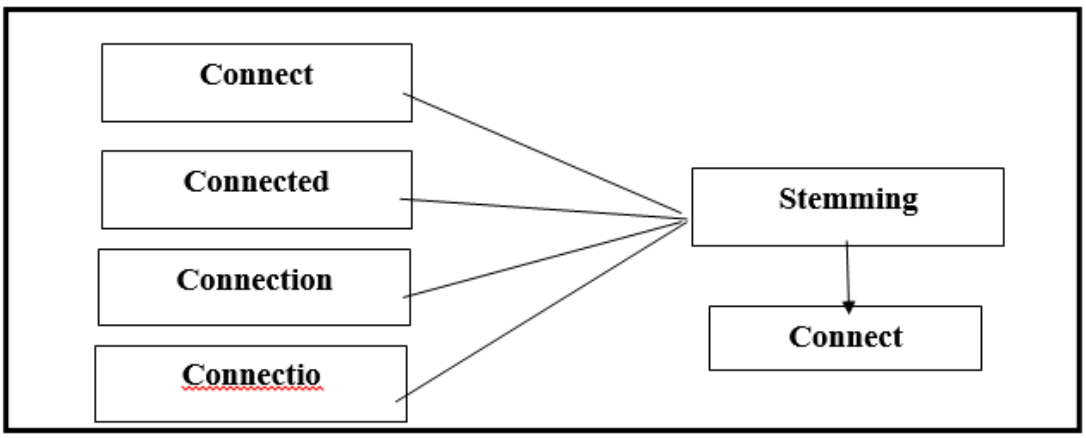

Figure 5.8 Types of words for stemming

Porters Stemmer Porters stemming algorithm [11, 12] is one of the most popular stemming algorithm proposed in 1980. Many modifications and enhancements have been made and suggested on the basic algorithm. It is based on the idea that the suffixes in the English language (approximately 1200) are mostly made up of a grouping of smaller and simpler suffixes. It has five steps, and within each step, rules are applied until one of them passes the conditions. If a rule is accepted, the suffix is

removed consequently, and the next step is performed. The resultant stem at the end of the fifth step is returned.

```
>>>
>>> text="Yep, they all wanted to be with us in our cool http://facebook.com car, it is fun. Right @maetzju?
>>> textWithOutLink=removelink(text)
>>> textWithOutLink
'Yep, they all wanted to be with us in our cool  car, it is fun. Right @maetzju?  #BreakDancer #Springfest'
>>> TextWithOutPuncAndStopWord=text_procss(textWithOutLink)
>>> TextWithOutPuncAndStopWord
'Yep wanted cool car fun Right maetzju BreakDancer Springfest'
>>> StemText=stemming(TextWithOutPuncAndStopWord)
>>> StemText
' yep want cool car fun right maetzju breakdanc springfest'
>>>
```

Figure 5.9 Screenshot of stemming process of data set

5.3.7 Case folding

Users do not bother about how they type the words like the small letter, capital letter or title letter. Same here in our dataset some user type the hashtag in capital letter, or they differentiate the two words that write together by capitalization of the first character of each word.

i.e. #SpringFest, #RamadanMonth. Therefore our comparisons are case-insensitive.

So we normalized the text for comparison, for this purpose we did case folding, where we convert all terms to lower case and then we did case insensitive compassion.

5.3.8 TF-IDF

TF stands for term frequency and IDF stands for inverse document frequency. TF is the frequency of a term in a corpus and IDF is the ratio of log N/n. N is the total number of documents in the corpus and small n is the total number of documents have that term.tf-idf is the cross product of TF and IDF and its numerical statistic use to determine the importance of a term in the corpus. mostly use a weighting factor in information retrieval, text mining techniques, and user modeling. The tf-idf scoring for term *i* increases when the number of documents increase that have term *i*. The formula is as follow:

$$Tf - Idf_i = Tf_i x\ log^{\left(\frac{N}{n}\right)}$$

5.3.9 Feature Extraction and Graph Generation

To generate a graph for tweets and represent tweets with the attributed graph, we consider some features of tweets to form the graph. As follow:

1. Hashtags overlapping of tweets pairs
2. Top Words in corpus

Hashtags mostly contain the most important words for detecting the subjects of a tweet. We use hashtags overlapping between two tweets to add weighted edges or link. Weight is the number of common hashtags between tweet pairs.

To do this, we create a list of hashtags called it hash list for each tweet extraction of hashtags from our tweet dataset before doing the pre-processing of the dataset. Then we take the intersection between each two tweet pair hash list and count the number of hashtags in the result of intersection then used that number as the weight of links between the corresponding tweet pair.

```
>>> hashlist
[['breakdancer', 'springfest'], ['springfest'], ['umes', 'springfest'], ['springfest'], ['springfest'], ['springfest'], ['sprin
'], ['springfest'], ['springfest', 'laketravis'], ['afghanistan', 'kabul', 'kabulblast'], ['afghanistankandahar', 'kabul'], ['a
nistan', 'heartbreaking'], ['afghanistan'], ['afghanistan'], ['afghanistan'], ['afghanistan'], ['afghanistans'], ['kabul', 'afg
tahaffuzmovement'], ['pashtuntahaffuzmovement', 'pashtunlongmarch2swat'], ['pashtunlongmarch2swat', 'pashtuntahaffuzmovement'],
ement'], ['pashtuntahaffuzmovement'], ['pashtunlongmarch2swat', 'pashtuntahaffuzmovement'], ['pashtunlongmarch2swat', 'pashtunl
longmarch2swat', 'pashtuntahaffuzmovement'], ['pashtunlongmarch2lahore', 'pashtuntahaffuzmovement', 'pashtunrejectstateterroris
ch2swat', 'ptm', 'pashtuntahaffuzmovement'], ['pashtuntahaffuzmovement'], ['pashtuntahaffuzmovement', 'pashtunlongmarch2swat'],
wat', 'pashtuntahaffuzmovement'], ['pashtuntahaffuzmovement'], ['pashtuntahaffuzmovement'], ['pashtunlongmarch2swat', 'pashtunt
pashtunlongmarch2swat', 'pashtunlongmarch'], ['pashtunlongmarch2swat', 'pashtuntahaffuzmovement'], ['pashtunlongmarch2lahore',
ent', 'pashtunrejectstateterrorism'], ['pashtunlongmarch2swat', 'ptm', 'pashtuntahaffuzmovement'], ['ramadanmonth', 'fasting',
n', 'sheratonabuja', 'hotel', 'ramadan', 'vouchers', 'gift'], ['ramadanmonth', 'adegaramadan', 'blessings'], ['ramadanmonth', '
ings'], ['ramadanmonth'], ['aldiarhotels', 'ramadanmonth'], ['ramadanmonth'], ['sosad', 'turkey', 'bangladesh', 'ramadanmonth']
h', 'wideawake']]
>>>
```

Figure 5.10 List of Hashtags of the data set

We assume each tweet as a node for generation of the attributed graph. and 1 × 50 binary array form the nodes also we can call it attributes.for this purpose we have maintained the list of top 50 words extracted from our dataset base on tf-idf scoring and then we have used these top words for adding value to the attribute of nodes, as in order, if top word present in tweet we put 1 else zero.

```
>>> topwords
['peac', 'best', 'taliban', 'face', 'slap', 'statepashtunlongmarch2swat', 'stori', 'may', 'adegaramadan', 'attend', 'despit
housand', 'world', 'famili', 'get', 'miss', 'pashtunlongmarch', 'persons', 'wish', 'pakistan', 'kabul', 'jalsa', 'shereenya
'everyth', 'day', 'fast', 'manzoor', 'pashtoon', 'journalist', 'stop', 'afghan', 'heartbreak', 'readi', 'thank', 'kill', 't
fun', 'attack', 'afghanistan', 'peopl', 'ramadanmonth', 'springfest', 'pashtunlongmarch2swat', 'pashtuntahaffuzmov']
>>>
```

Figure 5.11 List of top-words

5.3.10 Simple Graph Generation for Multi-attributed Graph

As far after doing above all the site. We get a multi-attributed graph, where nodes have 50 binary attributes and there are weighted edges between vertex pairs based on hashtag overlapping. Now we use the proposed method to convert this attributed graph into the simple graph, as follow.

1. MAG-Dist
2. MAG-Sim

The simple graph will look as follow:

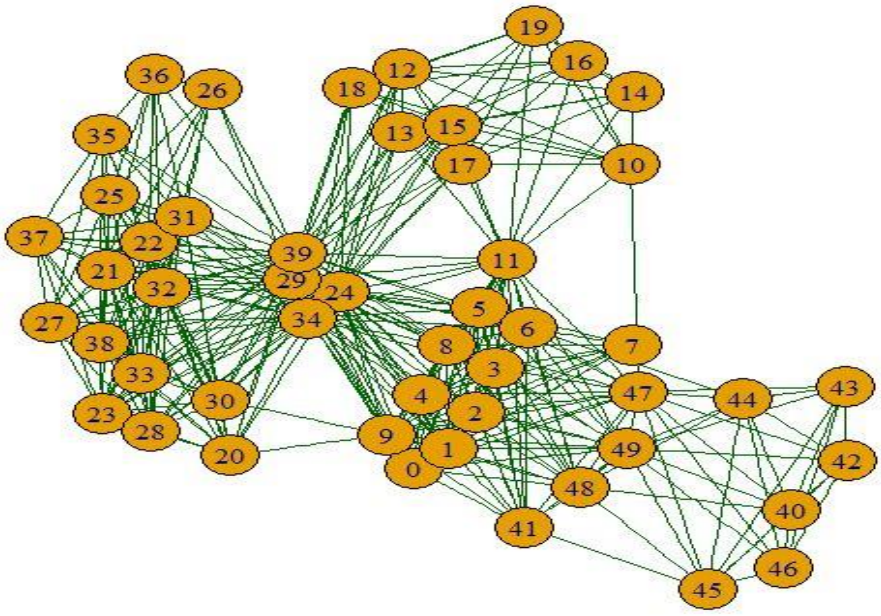

Figure 5.12 Simple graph after Distance measure of generated graph

Now we use MCL (Markov clustering) and we will get the result as follow:

Table 5-5 Clustering results

Clusters	SpringFest	Afghanistan News	PashtunTahafuzLongMarch	RamadanMonth
C1	9	0	0	0
C2	0	7	0	0
C3	0	0	20	0
C4	0	0	0	7

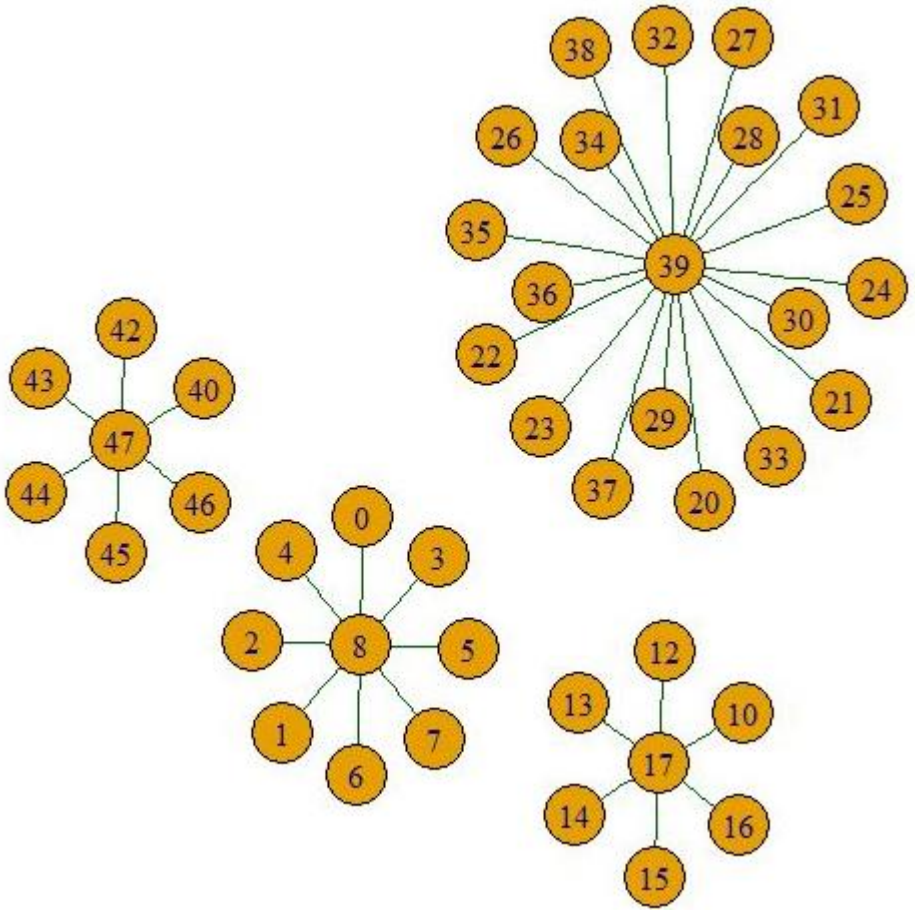

Figure 5.13 Clustered tweets graph

After evolution of the result the accuracy is as follow:

Table 5-6 Accuracy of generated clusters

Clusters	Efficiency rate
C1	0.9
C2	0.7
C3	1
C4	0.7
Average	0.825

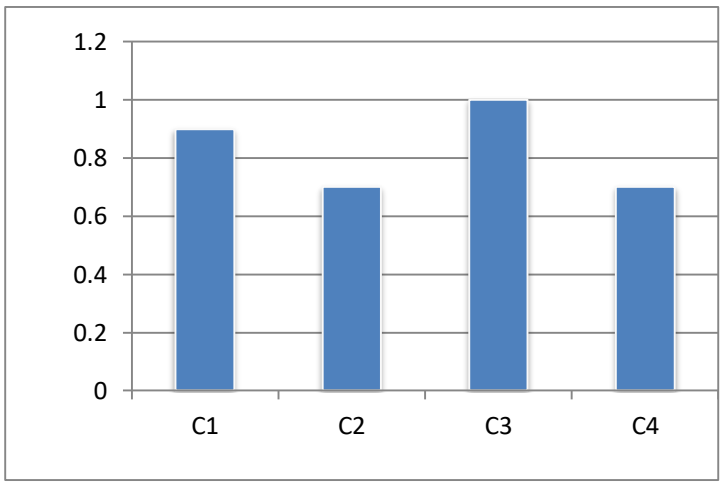

Figure 5.14 Visualization of accuracy values

Chapter 6

Conclusion

The growth in social networks results in the big amount of rich-data with information about users, society and social behaviors of users in a social network. To give a good experience to the users in social networks and extract this rich information from the social network's data, which can be further used for various useful purposes. It is recommended to decompose the large social networks to small communities which will ease the analysis and understanding procedure of a large social network. Thus in this thesis work, we have used approaches and methods for generation and analysis of social networks, which can be represented as the multi attributed graph. And these approaches yield suitable clustering or community results. So with the help of these methods, we can analyze different social networks which are form as a multi-attributed graph. Here I realized that the main focus should be on graph generation as a multi-attributed graph. So for this, we have to be careful in the selection of features in data, features are needed for social graph generation, and the efficiency increases long we use more and proper features from the data. The efficiency of used approaches is evaluated after analysis of standard and well-known iris dataset. The used approaches in my experiments yield better result compare to many other available approaches for the same kind of problem-solving.

References

[1] Campbell, William M., Charlie K. Dagli, and Clifford J. Weinstein. "Social network analysis with content and graphs." *Lincoln Laboratory Journal* 20.1 (2013): 61-81.

[2] Basu-Roy, Senjuti, Tina Eliassi-Rad, and Spiros Papadimitriou. "Fast and Effective Pattern Matching on Weighted Attributed Graphs." *ACM Knowledge Discovery and Data Mining* (2013).

[3] Gallagher, Brian. "Matching structure and semantics: A survey on graph-based pattern matching." *AAAI FS* 6 (2006): 45-53.

[4] Zhou, Yang, Hong Cheng, and Jeffrey Xu Yu. "Graph clustering based on structural/attribute similarities."*Proceedings of the VLDB Endowment* 2.1 (2009): 718-729.

[5] Agrawal R, Gehrke JE, Gunopulos D, Raghavan P, inventors; International Business Machines Corp, assignee. Automatic subspace clustering of high dimensional data for data mining applications. United States patent US 6,003,029. 1999 Dec 14.

[6] Muhammad Abulaish and Jahiruddin, "A Novel Weighted Distance Measure for Multi-Attributed Graph", In Proceedings of the 10th Annual ACM India Conference (COMPUTE), Bhopal, India, Nov. 16-18, 2017.

[7] Mohammed J. Zaki and Wagner Meira Jr. 2014. "Data Mining and Analysis: Fundamental Concepts and Algorithms". Cambridge University Press, New York, USA.

[8] Peter J. Olver. 2008. "Numerical Analysis Lecture Note". Retrieved on 18.03.2017from http://www-users.math.umn.edu/~olver/num_/lnn.pdf

[9] "Neo4j Operation Manual" https://neo4j.com/docs/operations-manual/current/introduction/.

[10] "Introduction to PageRank" http://www.amsi.org.au/teacher_modules/pdfs/Maths_delivers/Encryption5a.pdf.

[11] Vijayarani, S., Ms J. Ilamathi, and Ms Nithya. "Preprocessing techniques for text mining-an overview." International Journal of Computer Science & Communication Networks 5.1 (2015): 7-16.

[12] Porter, Martin F. "An algorithm for suffix stripping." *Program*14.3 (1980): 130-137.

[13] Porter, Martin F. "Snowball: A language for stemming algorithms." (2001).

[14] Bhat, Sajid Yousuf, and Muhammad Abulaish. "A Unified Framework for Community Structure Analysis in Dynamic Social Networks." *Hybrid Intelligence for Social Networks*. Springer, Cham, 2017. 77-97.

[15] Fazil, Mohd, and Muhammad Abulaish. "Why a socialbot is effective in Twitter? A statistical insight." *Communication Systems and Networks (COMSNETS), 2017 9th International Conference on*. IEEE, 2017.

[16] Ng, Raymond T., and Jiawei Han. "E cient and E ective Clustering Methods for Spatial Data Mining." *Proceedings of VLDB*. 1994.

[17] Schaeffer, Satu Elisa. "Graph clustering." *Computer science review* 1.1 (2007): 27-64.

[18] Cheng, Jiefeng, et al. "Fast graph pattern matching." *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*. IEEE, 2008.

[19] Vlasblom, James, and Shoshana J. Wodak. "Markov clustering versus affinity propagation for the partitioning of protein interaction graphs." *BMC bioinformatics* 10.1 (2009): 99.

[20] Kwak, Haewoon, et al. "What is Twitter, a social network or a news media?." *Proceedings of the 19th international conference on World wide web*. ACM, 2010.

[21] Yang, Shengqi, et al. "Fast top-k search in knowledge graphs." *Data Engineering (ICDE), 2016 IEEE 32nd International Conference on*. IEEE, 2016.

[22] Safaei, Marjaneh, Merve Sahan, and Mustafa Ilkan. "Social graph generation & forecasting using social network mining." *Computer Software and Applications Conference, 2009. COMPSAC'09. 33rd Annual IEEE International*. Vol. 2. IEEE, 2009.

[23] Gromann, Dagmar, and Thierry Declerck. "Hashtag Processing for Enhanced Clustering of Tweets." *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*. 2017.

[24] Song, Yu-Chen, et al. "Applications of attributes weighting in data mining." *IEEE Proc. of SMC UK &RI 6th Conference on Cybernetic Systems*. 2007.

[25] Narayanan, Srinivas, Nandagopal Venkataramanan, and Eric Sun. "Automatically generating nodes and edges in an integrated social graph." U.S. Patent No. 8,185,558. 22 May 2012.