

EXPLORING AREA-SPECIFIC MICROBLOGGER SOCIAL NETWORKS

by

Ece Aksu Değirmencioğlu

B.S., Computer Engineering, Ege University, 2002

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering
Boğaziçi University

2010

EXPLORING AREA-SPECIFIC MICROBLOGGER SOCIAL NETWORKS

APPROVED BY:

Dr. Suzan Üsküdarlı

(Thesis Supervisor)

Assist. Prof. Haluk Bingöl

Assoc. Prof. Yağmur Denizhan

DATE OF APPROVAL: 20.04.2010

ACKNOWLEDGEMENTS

I would like to thank to my supervisor Dr. Suzan Üsküdarlı for her endless support and guidance throughout this thesis. My sincere appreciation goes out to all my committee members, Assist. Prof. Haluk Bingöl and Assoc. Prof. Yağmur Denizhan. I would also like to thank all my friends in the Complex Networks Research Lab – Soslab for their support and suggestions.

I am deeply grateful to my dear husband Çağdaş Değirmencioğlu for his patience, understanding and morale support to help me complete this thesis.

Finally I am very thankful to my colleagues at The Royal Bank of Scotland N.V for their understanding and support.

ABSTRACT

EXPLORING AREA-SPECIFIC MICROBLOGGER SOCIAL NETWORKS

Social networks can be used to find people who share similar interests or people who have knowledge in a specific domain. Using social networks to share knowledge is a very efficient way of reaching information. Current social networking tools provide many ways to search people with similar interests. However, they are either based on keyword search or ranking users based on popularity. Keyword search is limited to information explicitly declared by users such as name, location, marital status, interests etc. Since users often do not declare their interest areas or the content they contribute is not aligned with the area of interest they declare, it is usually a time consuming task to locate those who are of interest. User ranking methods, on the other hand, hides users who provide valuable information but not so popular. In this study we propose a model for determining the area of interests of users based on their contributions. In other words, we examine what they contribute rather than what they declare about themselves. The idea is that their value depends on what they contribute. Areas of interests are determined based on the co-occurrence of related words in user contributions. In addition, we explore communities of different interests, based on the common context different people use in their contributions. In order to put some semantic grounding to what we have found as interest areas, we map the content we extracted from the users' contributions to other resources such as DBpedia[84], Wikipedia[82] and Google[83]. We show that interest areas of people can be extracted from the dynamic content they provide. Besides, common interest networks of users can be generated by implementing our model. Furthermore, we can also generate networks of words which provide us a way to put semantics into the search queries instead of solely keyword based inquiries.

ÖZET

EXPLORING AREA-SPECIFIC MICROBLOGGER SOCIAL NETWORKS

Sosyal ağlar benzer ilgi alanlarına sahip olan yada belirli bir alanda bilgi sahibi olan kişileri bulmak için kullanılabilmektedir. Günümüzde sosyal ağların bilgi paylaşımı amacıyla kullanılması, doğru bilgiye verimli şekilde ulaşmak yada temel olarak sorularımıza doğrudan yanıtlar bulmak için kullanılabilecek bir yöntemdir. Mevcut sosyal ağlar üzerinde benzer kişileri yada belirli bir bilgiye sahip doğru kişileri bulmak oldukça zordur. Bu sistemler üzerindeki kişi arama yada benzer kişileri bulma uygulamaları ya kullanıcıların kendi verdikleri kişisel bilgilere dayanmakta yada kullandıkları bir kelime ile metin bazlı eşleştirme yaparak çalışmaktadır. Kullanıcılar sosyal ağlarda kendilerine ait tüm bilgileri paylaşmayabilmekte, ya da kendileri hakkında yanlış bilgiler verebilmektedirler. Kullanıcıların paylaştıkları içeriği değerlendirerek kullanıcılar arasında benzerlik bulmaya çalışan uygulamalar henüz gerçekleştirilmemiş durumdadır.

Bu çalışma kapsamında, sosyal ağ uygulamalarının bir türü olan mikroblogger üzerinde, kullanıcıların sağladıkları içeriği değerlendirerek sahip oldukları bilginin yada ilgilendikleri konuların hangi ilgi alanlarına ait olduğunu tespit etmek, bunun ötesinde diğer kullanıcılarla ortak ilgi alanlarına göre ilişki kurmak amaçlı bir model önerilmektedir. İçerik olarak kullanılan metin cümlecikleri içinde birlikte geçen kelimeler baz alınarak bu kelimeler arasında anlamsal bütünlük oluşturulmaya çalışılmakta, böylece kişilerin kendileri kullanmadıkları halde ilgi alanları ile ilişkili olabilecek olan diğer kelimeler tespit edilmektedir.

Çalışmamızın sonucunda kullanıcıların sosyal ağlarda paylaştıkları içerik dikkate alınarak ortak ilgi alanlarına sahip kullanıcılar arasında bağlantılar kurulabileceği, ayrıca kişiler kullanmamış olsa bile bu ilgi alanları ile ilgili diğer sözcüklerin de tespit edilebildiği gösterilmiştir.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	III
ABSTRACT.....	IV
ÖZET.....	V
LIST OF FIGURES.....	VIII
LIST OF TABLES	X
LIST OF SYMBOLS/ABBREVIATIONS.....	XI
1. INTRODUCTION.....	1
1.1. Motivation.....	1
1.2. Outline	2
2. BACKGROUND	4
2.1. Social Web and User Generated Content	4
2.2. Tag, Tagging And Collaborative Tagging Systems	9
2.3. Microblogging and Twitter.....	13
2.4. Social Network Analysis Basics	19
2.5. DBpedia, Wikipedia, Google.....	21
2.6. Related Work	22
3. PROBLEM STATEMENT.....	25
3.1. Sample Case.....	27
4. ANALYSIS.....	31
4.1. Understanding Twitter.....	31
4.2. Understanding User Behaviors	34
4.3. Understanding Relations between Users of a Community	36
4.4. Understanding the Tweets from Single User Perspective	38
4.5. Summary.....	41
5. PROPOSED MODEL	42
5.1. Processing Microblogs	46

5.2. Networks Analysis	52
5.2.1. Related Words Network.....	54
5.2.2. Users – Tags Network.....	58
5.2.3. Users – Tags – Words Networks	60
5.2.4. Interest Based User Networks	62
5.3. Semantic Grounding	64
6. IMPLEMENTATION	67
6.1. Implementation Platform.....	67
6.2. Twitter API	68
6.3. Implemented Functions	72
6.4. Pajek	78
6.5. Database.....	80
7. EVALUATION.....	83
8. CONCLUSION.....	94
8.1. Overview.....	95
8.2. Contributions.....	96
8.3. Future Work.....	96
APPENDIX A. STOP WORDS IN ENGLISH	99
APPENDIX B. INTEREST BASED USERS NETWORKS RESULTS.....	102
APPENDIX C. RESULTS SUMMARY TABLES FOR INTEREST BASED USER NETWORKS	103
REFERENCES.....	107

LIST OF FIGURES

Figure 2.1. Sample Social Network Diagram.....	4
Figure 2.2. Comparison of Web 1.0 and Web 2.0 [5].....	7
Figure 2.3. Main activities related to social media platforms [24].....	8
Figure 2.4. A Sample tag cloud for the Web 2.0 [5].....	11
Figure 2.5. Triple structure of tagging.....	12
Figure 2.6. Traffic on Twitter on hourly basis[29].....	15
Figure 2.7. Screenshot from Twitter.....	17
Figure 2.8. Spread of retweets.....	18
Figure 4.1. Twitter search [65].....	32
Figure 5.1. Proposed model.....	44
Figure 5.2. Processing microblogs algorithm.....	47
Figure 5.3. Common tag usage by different microbloggers.....	49
Figure 5.4. Sample Related Words Network.....	54
Figure 5.5. Generating Related Words Network algorithm.....	55
Figure 5.6. Sample Related Words Network.....	56
Figure 5.7. Sample Users-Tags Network.....	58
Figure 5.8. Sample Users-Tags Network.....	60
Figure 5.9. Sample Users-Tags-Words Network.....	61
Figure 5.10. Sample Interest Based Users Network.....	62
Figure 5.11. Users Network for a sample user.....	64
Figure 6.1. Source code to retrieve tweets using Twitter API.....	71
Figure 6.2. Source code for finding replied users and processing their tweets.....	75
Figure 6.3. Source code for a sample SPARQL query for the word “iPhone”.....	76
Figure 6.4. Source code for API call to DBPedia.....	76

Figure 6.5. Source code for API call to Wikipedia.....	76
Figure 6.6. Source code for API call to Google.....	76
Figure 6.7. Sample Pajek input data.....	80
Figure 7.1. Connected and isolated users.....	85
Figure 7.2 Interest Based User Network for a seed user who is not a central node.....	86
Figure 7.3. Central nodes of the seed user given in Figure 7.2.....	86
Figure 7.4 The ratios of central nodes/all nodes in all networks.....	87
Figure 7.5. Central nodes in an interest based users network with 3 central nodes.....	89
Figure 7.6. Overall interest based users network with 18 nodes(3 central).....	89
Figure 7.7. Central nodes in an interest based users network with 16 of central nodes.....	90
Figure 7.8. Overall interest based users network with 20 nodes(16 central).....	90
Figure 7.9. Associated tags and words in a sample user network with 3 central nodes.....	91
Figure 7.10. Associated tags and words in a sample user network with 16 central nodes..	92
Figure C.1 The ratio of central nodes to all nodes in each interest based user network...	103
Figure C.2 The ratio of central nodes to all nodes in each interest based user network....	104
Figure C.3 Summary of average degrees.....	104
Figure C.4 Summary of average closeness values.....	105
Figure C.5 Summary of average betweenness values.....	106

LIST OF TABLES

Table 2.1. Sample tweets.....	15
Table 2.2. Sample tiny URL.....	16
Table 2.3. Sample SPARQL query.....	21
Table 3.1. Sample users.....	28
Table 4.1. Data for the experiment – understanding Twitter.....	32
Table 4.2. Swine flu tweets.....	33
Table 4.3. Results for the experiment – understanding Twitter.....	34
Table 4.4. Data for the experiment – understanding user behaviors.....	35
Table 4.5. Results for the experiment – understanding user behaviors.....	35
Table 4.6. Data for the experiment – relations between users of a community.....	37
Table 4.7. Results for the experiment – relations between users of a community.....	37
Table 4.8. Data for the experiment – understanding from single user perspective.....	39
Table 4.9. Results for the experiment – tags of a sample user.....	39
Table 4.10. Results for the experiment – words of a sample user.....	40
Table 6.1. DBpedia result set.....	77
Table 7.1. Properties of data set in terms of frequencies.....	83
Table 7.2. Evaluation of data.....	84
Table 7.3. Network measures comparison for two sample networks.....	87
Table A.1. Stop words in English.....	99
Table B.1. Results for each of the Interest Based Users networks for 49 seed users.....	102

LIST OF SYMBOLS/ABBREVIATIONS

API	Application Programming Interface
HTTP	Hypertext Transfer Protocol
REST	Representational State Transfer
RT	Re-Tweet
UCC	User Created Content
UGC	User Generated Content
URL	Uniform Resource Locator
WWW	World Wide Web
XML	Extensible Markup Language

LIST OF TERMS AND CONCEPTS

Related Words Network	A network of words and tags where words are connected to tags based on co-occurrence relations and all connections are weighted by the frequency of their co-occurrence.
Users – Tags Network	Network of users and tags where users are connected to the tags they use.
Users – Tags – Words Network	Network of words, tags and users where users are connected to the tags they co-occur with. It is a combination of users – tags network and the related words network
Interest Based User Network	Network of users where users are connected to each other if they use common tags. The connection is weighted by the number of common tags they use.

1. INTRODUCTION

1.1. Motivation

Social applications in the Internet today allow people interactively share their knowledge, resources such as video, text and images, collaboratively perform tasks such as updating documents, find friends and communicate. The popularity of these applications in online social networks increased after the advent of Web 2.0 [5] technologies and user generated content [80]. Among different types of social applications, microblogging environments became the fastest growing type of applications with the introduction of Twitter in 2007. With its over 14 million users, Twitter's growth has been declared as 1392 percent in 2009.

Microblogging provides a very simple, short but efficient and quick way of spreading and retrieving information. It allows users to expose their ideas, feelings, interests, knowledge and expertise by means of short text messages. Users also interact through microblogs. To express the desired content in a space efficient manner, various space conserving conventions and notations have emerged. For example in Twitter, hashtags are tokens that start with a hash symbol (#) prefix. They are used to tag microblogs (tweets) they occur in. A microblogger's contributions:

- may relate to numerous topics
- are fragmented thoughts into many microblogs
- may be duplication of somebody else's content
- may contain references to other users, external links, tags etc.

The nature of microblogs causes two major problems. First, it is very difficult to distinguish valuable content among all the contributions. Second, it is hard to find users who contribute actively and contribute valuable information to follow.

Search tools for content and users are available for microblogging environments and specifically for Twitter. One of the two approaches is keyword matching. People are encouraged to search for specific keywords to apply keyword matching to find information

which contains the given keywords. Keyword matching is used either to find content [65] or users [38]. When content is searched by keywords, it causes thousands of contributions from many different users to appear in the result set which make it very hard to filter out the irrelevant content. When users are searched by keywords, the result set depends on the explicit information declared by users themselves. However, in the case that people's declarations and their contributions are not aligned, misleading results are returned from queries by specific keywords. The other approach is popularity based ranking. With this approach, discovery of users who share relevant and valuable information is based on the popularity of users measured by quantitative variables such as the number of followers or contributions. This kind of search tools, leave the less popular but more valuable users hidden among millions of other users.

In this study, we focus on finding users who are interested in a specific area by processing their microblog contributions. We also focus on finding the relations between users who build up a community around similar interests. This work aims to;

- (i) identify a microblogger's interest given their contributions
- (ii) identify a community of interest given a microblogger
- (iii) examine social network properties of microblogger communities of interest

It proposes first to process a collection of microblog contributions and reduce them into a set of keywords representing the nature of their content. Secondly, to identify a community of interest based on a user.

1.2. Outline

In the following chapter, background information related to our work is given in detail. First, basic concepts of social web and user generated content (UGC) are described. In addition, main issues regarding the user generated content is explained with references to the related work in this area. Next, tagging and collaborative tagging behavior of users in social networks is described briefly as a solution to solve the issues regarding the UGC. Then we describe the concept of keyword co-occurrence which we utilized in our method in order to find the related words in a given interest area. Finally we explain microblogging

and Twitter environment which we chose as test bed for our study in this research. In the next chapter, we also give brief information about Social Network Analysis.

In Chapter 3, we define our problem statement together with a sample scenario regarding the problem.

In Chapter 4, the results of our analysis which has been performed to understand the structure of social networks, how people behave in microblogging environments and characteristics of Twitter environment is described.

In Chapter 5, we propose our model to explore interest area specific users and communities in microblogging environments.

In Chapter 6, we briefly present the implementation of our model that has been completed as part of this research.

In Chapter 7, we present the cases we have tested to evaluate our model together with the results for each case.

In Chapter 8, we discuss our model to provide information about the constraints and limitations identified during evaluation phase. In addition, we provide detailed information about the future work and a summary of our research and our contributions as conclusion of our work.

2. BACKGROUND

In this chapter we provide an overview of social networks, user generated content and enabling technologies for them. We first describe social web and online social networks in Section 2.1 and explain how they are used in today's World Wide Web. In this section, we also explain enabling technologies for social websites where users can generate content. In section 2.2, we provide detailed information about the bottom up classification approach, which is called tagging, to categorize the content that users generate. In Section 2.3, microblogging concept and a sample microblogging environment Twitter is explained since our research is mainly based on the text content provided by the users via microblogging web sites. In Section 2.4, we define basic terms and concepts in the area of social network analysis which we applied to evaluate the results of our implementation. In section 2.5, we give a brief overview of the web sites DBpedia [84], Wikipedia [82] and Google [83]. Finally we provide information about the research related to our work.

2.1. Social Web and User Generated Content

The relationships between individuals and groups of individuals in a community are represented by social networks in terms of nodes and edges where nodes are the individuals and the edges are the relationships between the individuals [1][2].

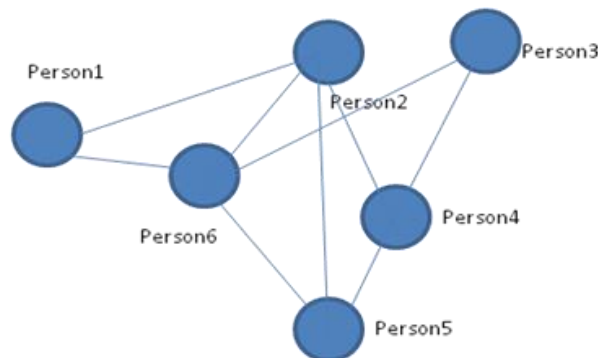


Figure 2.1. Sample social network diagram

Internet today allows people create virtual social networks by means of web applications [1]. People communicate, share information, find friends and connect other

people who have similar interests throughout World Wide Web. The term social web describes these new ways of socializing and interacting on the web [3]. In other words, it can be described as people interlinked and interacting with engaging content in a conversational and participatory manner via the Internet [4]. Main attributes of a social web application are listed as [3]:

- Identity: who the person is
- Reputation: what do people think the person stands for
- Presence: where the person is from
- Relationships: who is the person connected with? who does the person trust?
- Groups: how does the person organize their connections?
- Conversations: what does the person discuss with others?
- Sharing: what content does the user make available for others to interact with?

We see some or all of the characteristics of social web applications mentioned above in social web sites Facebook [63], Twitter [59], FriendFeed [60] and many others.

There are two types of interaction between people on the social web. People may communicate directly through social applications and web sites such as Facebook [63], Bebo [55], Myspace [56] or they may interact indirectly by sharing content in a participatory manner through applications or websites such as Flickr [57], Del.icio.us [58], Twitter [59], FriendFeed [60], DeviantArt [61]. In our study, we focus on the second type of social networks where people communicate through an interest.

The enabling technologies of Social Web applications are associated with the term Web 2.0 which is presented in O'Reilly Media Web 2.0 conference in 2004 by Tim O'Reilly [5]. Web 2.0 technologies allow people publish content, share knowledge and interact with other people in contrast to the Web 1.0 technologies where users can only retrieve information in a passive manner. The main characteristics of the Web 2.0 technologies are that they use web as a platform and require building applications harness network effects to get better the more people use them [5].

Web 1.0 was all about the hypertext documents which are linked so that people could navigate from one document to another. The content was provided by the web sites and users had no opportunity to contribute to the content. The important issue was to create content or present the existing content in the form of hypertext.

Web 2.0 provided a way to develop web sites where people can interact, communicate and contribute to the content. Not only it offers content, but it also offers interoperable and re-usable development of services so that people can store, process and retrieve the content across computers and other devices such as mobile phones. For instance, Flickr allows users publish, comment, organize images; Del.icio.us store, share and access their bookmarks, Google Docs [62] collaboratively work on documents. In addition, the service based platform offered by Web 2.0 allows different applications to be combined and re-used in other web sites. Google Maps [64] is a good example of such combined applications which are called mash-up applications.

A brief comparison of the Web 1.0 and Web 2.0 is given in the Figure 2.2 [5].

Due to the rich and easy-to-use interfaces of Web 2.0 applications, the number of people using these services is increased enormously. Social Web applications built on top of the Web 2.0 technologies create a seamless platform for people to communicate and share knowledge.

Web 1.0		Web 2.0
DoubleClick	-->	Google AdSense
Ofoto	-->	Flickr
Akamai	-->	BitTorrent
mp3.com	-->	Napster
Britannica Online	-->	Wikipedia
personal websites	-->	blogging
Evite	-->	upcoming.org and EVDB
domain name speculation	-->	search engine optimization
page views	-->	cost per click
screen scraping	-->	web services
Publishing	-->	participation
content management systems	-->	wikis
directories (taxonomy)	-->	tagging ("folksonomy")
Stickiness	-->	Syndication

Figure 2.2. Comparison of Web 1.0 and Web 2.0 [5]

The content produced and published by people, so called end-users of the Web 2.0 applications, are defined as User Generated Content (UGC) or User Created Content (UCC) [6]. The content may be in any format of images, bookmarks, text, wikis, blogs, video etc. Main online activities related to the UGC are:

- Blogs: mashable [72], readwriteweb [73]
- Microblogs: Twitter [65],
- Social Networking Sites: Facebook [63], MySpace[56]
- Trip Planners: YahooTravel [74]
- Photos & Videos: Flickr [57]
- Bookmarking: Del.icio.us [58]
- Customer Review Sites: TripAdvisor [75], IMDB [76]

The key idea behind the UGC is that the content is not published and organized by central administrators or authorities. Instead, end users publish their own content and

comment on the content published by others. Characteristics of the UGC are defined in the study by OECD [6] as

- Publication requirement
- Creative effort
- Creation outside of professional routines and practices

UGC activities are tracked by the Universal McCann [24] since September 2006. The result of a survey in 29 countries including 17,000 internet users shows that there is an impressive increase in the use of all kinds of social platforms. As of March 2008, the list of activities and the increase of each of these activities are given in the Figure 2.3.

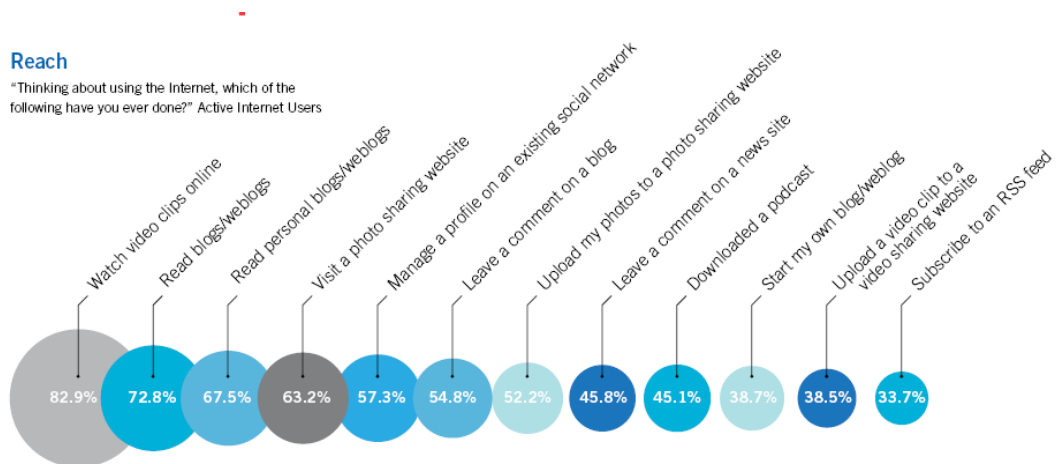


Figure 2.3. Main activities related to social media platforms [24]

UGC provides a wider content, less restrictive and easy to use functions to enter content and coverage of vast knowledge areas in a single platform. Besides, it provides experts who may not be the part of the web site development team to share their knowledge and users become a participant of the content.

However, there are some drawbacks of the UGC as well. Since the content is published by many users instead of a single administrator, it causes duplication and overlap of the content. The quality of the content is another issue about the UGC due that no restrictions or filtering are available for the published content. In addition, the credibility of

the user who publishes the content is not always apparent. Hence, the expertise and knowledge level of the users is questionable; therefore credibility of the content is also questionable in this respect. As a final note about the issues related to the UGC, we can say that the valuable data is hard to find among many other irrelevant and questionable content. This issue is directly associated with the non-structured format of the content which is published by users in free format and lack of organization and classification facilities for the UGC.

In this research we focus on the microblogs which contain text format of UGC and have a limitation of 140 characters in the text. We analyze the content in order to explore the interest areas of users and relations between other people based on the interest areas we extract from the content that the users publish. We use tags as a starting point to associate the users and their interests. In the following section, we explain the concept of tagging which is a bottom up approach to organize the UGC.

2.2. Tag, Tagging And Collaborative Tagging Systems

A tag is a non structured, informal and personal keyword or term which has no hierarchy definition and is assigned to a piece of information (such as an internet bookmark, digital image, or computer file) [78]. Tags are used as metadata [79] which helps describing and identifying the resource or information so that the search engines can retrieve it. A metadata is described as the “structured information that describes, explains, locates or otherwise makes it easier to retrieve, use or manage an information resource” [7]. A tag may be any kind of words or terms such as subject matter of the information, its name, location, reminder, personal note, feelings or phrases such as “to do” etc. Tags given to an information resource differ from user to user and they may be expressive or non-descriptive depending on the users’ perception and behavior [8].

Development of services associated with the Web 2.0 technologies provided the term tag to be popularized. The mainstream use of the tags started with the web site Del.icio.us [58]. Del.icio.us is a social bookmarking tool which allows users to add tags to their bookmarks.

As we explained in the preceding section, UGC causes some problems such as duplicate information, lack of organization, difficulties with searching and retrieving the data due to its informal and unstructured nature. Tags provide the functionality to help users organize and classify the content and also to mark the ownership and identity of the information. Tagging helps users find the information later [78].

Tag clouds are representations of all the tags used in a system or assigned to a resource or assigned by a user in a form that the most frequent tags can be distinguished in a single view. The size or the color of the more frequent words allow us identify them. A sample tag cloud for the Web 2.0 is given below [5].

Tags can be considered as a bottom up approach for classification when compared to the taxonomies which are defined by experts for a limited set of items hierarchically with a top down approach. In taxonomy, there is one way to classify each item. However, tags can be classified in many different ways since it has a flat structure [9].

A special type of tags is hashtag which is used in the microblogging systems such as Twitter [65] and Tumblr [71]. Hashtags are the words or phrases with prefix hash sign (#) and with multi words concatenated [78]. Throughout this document we refer to hashtags either as tag or hashtag.

Tagging, on the other hand, is the activity that users assign a tag to a resource or information either published by them or by other users. It has three elements: user, tag and resource. User is the person who tags the resource. They are named as taggers as well. Resource is the tagged item which can be any type of information such as image, text, bookmark, video, audio etc. Finally, tag is the keyword or term that the tagger assigns to a resource.



Figure 2.4. A sample tag cloud for the Web 2.0 [5]

There is no information regarding the meaning or semantics of a tag. While users freely enter tags, they classify the resources personally which makes it difficult for other users to search and retrieve the resources.

Users tag resources collaboratively in tagging systems such as Del.icio.us [58] and Flickr [57]. Collaborative tagging is described as the process by which many users add metadata in the form of keywords to shared content [9]. Growing number of social web sites allow their users tag not only their content but also the content published by others to organize them or make them searched and retrieved easily. The tags become a folksonomy when used collectively or collaboratively [11]. The term used by Mathes in 2004 as a combination of the words folk and taxonomy to reflect that it is a kind of classification created by people.

Collaborative tagging also offers alternatives to the on going effort in the area of semantic web ontologies [10]. Researchers proposed methods to model tagging activity [23][25][26][12]. Common structure of tagging is modeled with the three entities of the activity: user, tag, resource (See Figure 2.5).

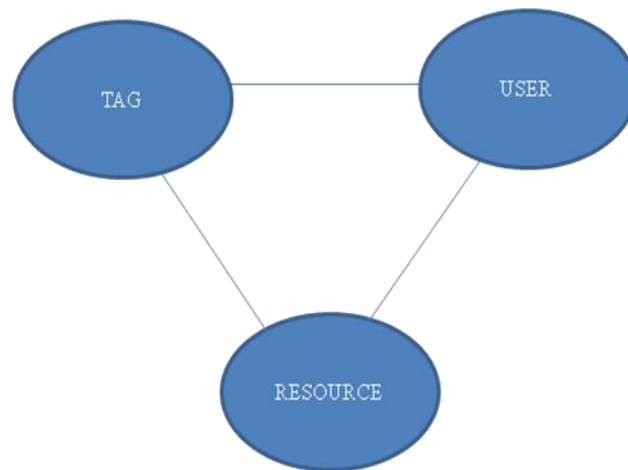


Figure 2.5. Triple structure of tagging

Gruber in 2005 [4], suggested an extension to the model as below.

$$\text{Tagging: (object, tag, tagger, source + or -)} \quad (2.1)$$

Object, tag and tagger correspond to the resource, tag and user in the previous model where source is used to filter the bad tags in order to avoid spammers. Gruber also introduced the role of tag ontologies which represent the tags as concepts and their relationship types [4]. Although the models define the tagging activity, they do not support collaborative tagging activity. Kim proposed another extension to the model in order to present a model for the collaborative tagging, namely folksonomy [12].

In our thesis, we refer to hashtags, tags with prefix hash sign (#), as tags throughout this document since they are specific use of tags in short messaging systems, so called microblogging systems, where people share text content. The microblogging system we have implemented our model is Twitter. The structure and dynamics of Twitter is explained in Section 2.3.

Twitter is a bit different from the collaborative tagging systems in terms of information resource type and the way people use it. A user publishes text content together with a hashtag attached in it. Other users retrieve and share this content with other users. In systems such as Flickr, users are able to add their own tags to the content which is

published by other users. However, in Twitter, adding tags by other users are limited due to the character limitation of text content which is 140 characters. This structure of Twitter makes our work different from the current research in the area of collaborative tagging. While collaborative tagging systems focus on discovery of resources tagged with a set of meaningful tags, we focus on discovery of users who tags resources which are text content in our case.

In the following section, microblogging environments and Twitter is explained in detail.

2.3. Microblogging and Twitter

Microblogging is a form of blogging that allows users to send brief text updates or other resources such as photos or audio clips and publish them, either to be viewed by anyone or by a restricted group which can be chosen by the user [53]. The messages can be published through web, mobile devices or desktop applications. People share news, give information about different areas of interest, share images, video or audio items, communicate person-to-person, provide comments or reviews, promote specific services or products, announce events or places etc. via microblogging. Microblogs are simpler short messages compared to the traditional blogging. Microblogs usually have a character limit for the text or size limit for the videos or images.

Microblogs gain popularity after the introduction of the services Twitter and Tumblr in 2006 and 2008. By May 2007, the number of microblogging sites was counted over a hundred in different languages [13]. Among them, Twitter, Tumblr, Plurk, Emote.in, Squeelr, Beeing, Jaiku and identi.ca are the ones we would like to mention here. Microblogging services influenced other social web sites such as Facebook, MySpace, LinkedIn[69] and XING[70] that they all adapted their system to provide microblogging services which are named as “status update” in these services.

As an additional note, research on microblogs has shown that the number of active users who create content or contribute is a small group when compared to the overall number of users [14][16]. A survey based on 11 millions of users shows that ten percent of

the users generate the 86 percent of the all activity in Twitter [15]. In our thesis, we consider this nature of microblogging services and we focus on exploring active users who publish valuable content in specific interest areas. One of our aims in this thesis is to connect people who have common interest which is very difficult to identify easily today among millions of other inactive users.

Twitter is free social networking and microblogging service which allows users to publish, share and retrieve content known as tweets. Twitter also supports video and image formats in addition to the text. The service was introduced in 2006 by Evan Williams and Jack Dorsay and significantly increased its usage by 2009. According to the market research [27], it is the fastest-growing web site for the February 2009 with the monthly growth of 1382 percent. Number of users has exceeded 14 million by April 2009 [27]. It is also ranked as one of the top 20 websites worldwide in Alexa's [28] web traffic analysis in January 2009. The number of updates is tracked by the service TweetSpeed [29] on hourly basis. Below is a screenshot taken from TweetSpeed on January 7, 2010. As it is shown in Figure 2.6., the number of updates per hour changes between 500,000 and 2 million depending on time.

Twitter is used for different purposes such as political campaigning, public relations, educational purposes, news, promotions, informative and conversational communication. For example, it was used in the 2008 U.S presidential campaign very actively by the candidate Obama.

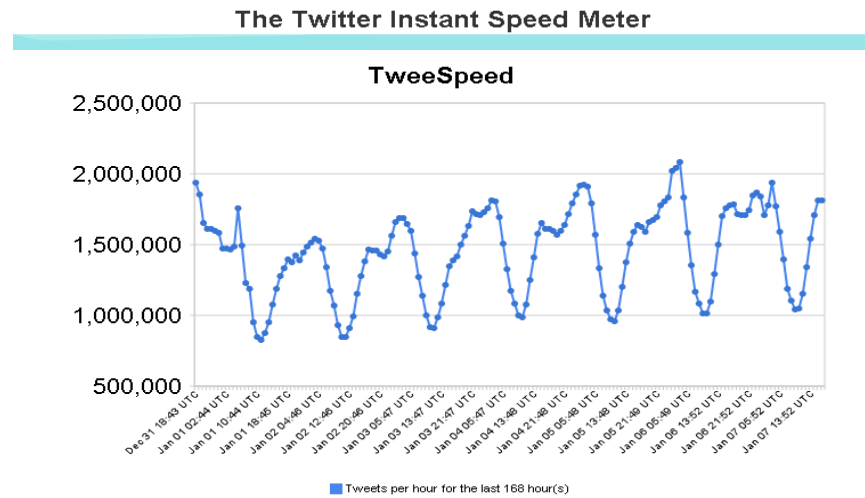


Figure 2.6. Traffic on Twitter on hourly basis [29]

Twitter has a character limit for each tweet which is restricted to 140 characters. The content of the tweets vary depending on the usage objectives of the users. Some samples are given below:

Table 2.1. Sample tweets

Sample Tweets
I am going to the gym and will be back by nine.
Free First Thursday today. Fellini film series starts tonight.

Similar to the short text messaging services (SMS), character limitation causes users to invent a new way of communication which is based on a short notation of words. The usage of tiny URL services is also increased since users avoid using long URL addresses. Tiny URL is a service that shortens the original URL address and redirects users to the original address. Some of these services are tinyURL [66], bit.ly [68] and goo.gl [67]. In our thesis, we refer to all the URL and tiny URL addresses we extract from the tweets as links.

On the other hand, short usage of text content forces users to give the main information precisely without using irrelevant words or indirect expressions.

Other than publishing tweets, users can follow other users in Twitter. Users are in follower role when they follow another user and listed in the followers list of the user they follow. The list of users who are being followed is named as friends. The tweets are broadcast to one direction in Twitter which means that the followers of a user are able to see the published tweets by that user. However, any other user who comes across with them in a search is also able to see any other user's tweets as long as the security and privacy settings are set to public instead of private option.

Table 2.2. Sample tiny URL

<u>Original Link</u>	<u>Tiny URL</u>
http://books.google.com.tr/books?id=QKB1AcdkMwsC&dq=Tagging:+People-Powered+Metadata+for+the+Social+Web&printsec=frontcover&source=bl&ots=HqVHN7S8I8&sig=oF9ENaC5anUQZgFAnPHGGECQiZE&hl=tr&ei=PMIFS4fgJNG04Qbt0cH0Ag&sa=X&oi=book_result&ct=result&resnum=5&ved=0CCkQ6AEwBA#v=onepage&q=&f=false	http://bit.ly/6L4op3

A reply is a special message sent from one user to another. It is distinguished from a normal tweet by the at sign (@) prefix of users. If a tweet begins with a @username, it is a reply. If the tweet has @username but not at the beginning of the tweet, it is considered as a mention in Twitter. Twitter displays the tweets which has @username on user's home page. There is no requirement for users to be following other users in order to see the replies or mentions to them. While replies and mentions are broadcast publicly to all other users who are not intended as well, Twitter also allows users to send private messages, named direct messages, from one person to another. Direct messages can only be sent to the followers. Direct messages are out of our scope in this study since they are not publicly retrievable.

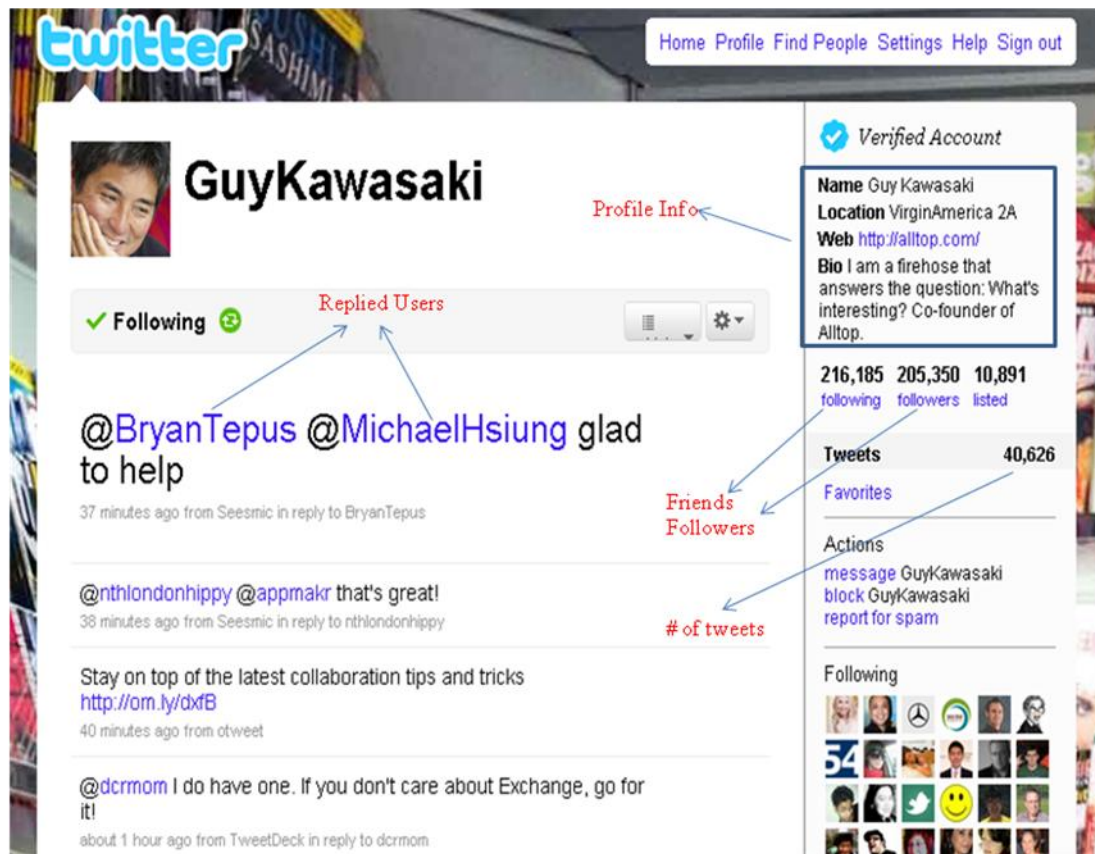


Figure 2.7. Screenshot from Twitter

ReTweet, in the social networking and micro-blogging service Twitter, to re-post something posted by another user, usually preceded with "RT" and "@username" to refer to the original poster. Retweets are used very frequently in twitter and has a dramatic influential effect on users [77]. Figure 2.8 shows the structure of spread of the retweets in a single view.

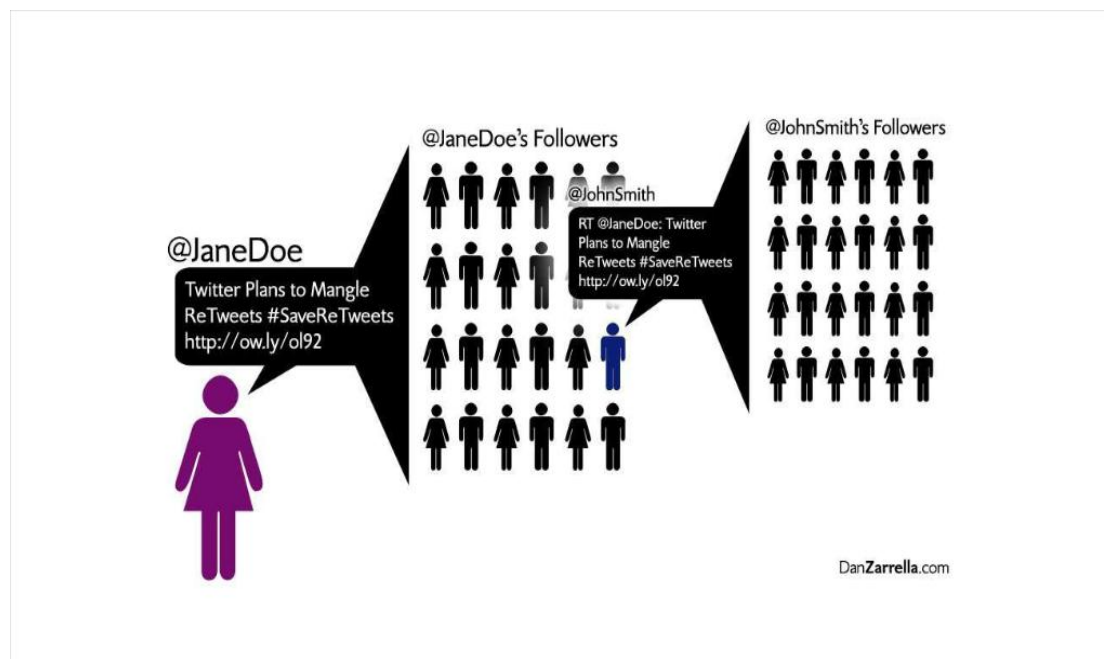


Figure 2.8. Spread of retweets

Users in Twitter are represented with a unique screen name. They are optionally enter their profile information such as name, biographic information, location and web site address.

In order to organize the users they follow, users create lists in Twitter. By means of lists people are grouped by specific subjects or interest areas. However, it does not always show us that a user listed in a list publish valuable content in the subject matter of the list. On the other hand, users who are not listed in any of the lists, since lists are optional, may be producing more valuable content than the users we retrieve via lists. In our thesis, we keep, all explicit categorizations given by the users, out of our scope and focus on the content to explore the communities or user groups that come together around a specific area of interest. For this reason, we do not use lists as a parameter to discover the relationships based on interests.

2.4. Social Network Analysis Basics

Social Networks represent the structure of the relationships between individuals. The individuals are represented as nodes in a social network. The nodes are connected by different types of interdependencies such as friendship, beliefs, knowledge, like, dislike etc. In our study we generate social networks which connect microbloggers by common interest areas they share.

Social network analysis views the relationships between individuals. The graph structure of the social networks usually shows the characteristics of complex networks in network theory [26]. In our research we focus on online social networks which emerge in a bottom up structure by the collaborative knowledge sharing by people. The research to analyze the properties of collaborative tagging systems, so called Folksonomies, has shown that such networks show complex network properties [26].

In social network analysis, there are some metrics to measure the properties of the networks. Centrality metrics among them are the ones we introduce in this thesis. Centrality of a node demonstrates the relative importance of a node among other nodes in the network. Knowing the central nodes, we can identify the users who have common interests with many of the others in the network. Besides, in our case, we evaluate our model by using centrality measures.

We have measured the three measures of centrality in this research: degree centrality, betweenness and closeness.

Degree centrality is defined as the number of connections that a node has. If the graph is directed, the connections from other nodes are named as in-degrees and the connections to other nodes are named as out-degrees. The networks we refer in this thesis are all undirected networks, hence we consider out-degrees since both in-degree and out-

degree measures are same for undirected networks. For a graph $G: = (V, E)$ with n vertices, the degree centrality $C_D(v)$ for vertex v is:

$$C_D(v) = \frac{\deg(v)}{n - 1} \quad (2.2)$$

Betweenness is the measure which a node lies between other nodes in the network. This measure takes into account the connectivity of the node's neighbors, giving a higher value for nodes which bridge clusters. The measure reflects the number of people who a person is connecting indirectly through their direct links [30]. Vertices that occur on many shortest paths between other vertices have higher betweenness than those that do not.

For a graph $G: = (V, E)$ with n vertices, the betweenness $C_B(v)$ for vertex v is:

$$C_B(v) = \sum_{\substack{s \neq v \neq t \in V \\ s \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (2.3)$$

where σ_{st} is the number of shortest paths from s to t , and $\sigma_{st}(v)$ is the number of shortest paths from s to t that pass through a vertex v

Closeness is the degree that an individual is near all other individuals in a network (directly or indirectly). The distance between nodes in a graph is the number of edges in a shortest path connecting them. It is also known as geodesic distance [30]. It is also defined as the mean geodesic distance (shortest path) between a vertex v and all other vertices reachable from it:

$$\frac{\sum_{t \in V \setminus v} d_G(v, t)}{n - 1} \quad (2.4)$$

where $(n \geq 2)$ is the size of the network's connectivity component V reachable from v . Closeness can be regarded as a measure of how long it will take information to spread from a given vertex to other reachable vertices in the network.

2.5. DBpedia, Wikipedia, Google

In our thesis we inquire DBpedia [32] concepts, Wikipedia[82] resources and Google [83] data to find an abstraction for area of interests. A set of related words are defined as area of interest. Related words are determined based on the co-occurrence of words in user contributions. Once a set of related words are determined, an abstract category, which can define an area of interest, is aimed to be found by mapping each word to these resources.

DBpedia is a project which extracts structured information from the information created as part of the Wikipedia project. DBpedia allows users to query relationships and properties associated with Wikipedia[82] resources, including links to other related datasets. DBpedia has been described by Tim Berners-Lee as one of the more famous parts of the Linked Data project [31]. DBpedia is inquired via a special query language SPARQL [33]. Below a sample SPARQL query is given:

Table 2.3. Sample SPARQL query

```
PREFIX dbprop: <http://dbpedia.org/property/>
PREFIX db: <http://dbpedia.org/resource/>
SELECT ?who ?work ?genre WHERE {
  db:Tokyo_Mew_Mew dbprop:illustrator ?who .
    ?work dbpprop:author ?who .
OPTIONAL { ?work dbpprop:genre ?genre } . }
```

In the case that a word can not be found in DBpedia, we search for the word in Wikipedia [82] and Google [83] in the same way we do for DBpedia. However, instead of SPARQL queries, they have different APIs and different call structures to query their data. Wikipedia and Google APIs are explained in Chapter 6 during implementation.

2.6. Related Work

Social networks can be used to find people who share similar interests or people who have knowledge in a specific domain. Using social networks to share knowledge is a very efficient way of reaching information or simply finding answers to questions.

There are two types of methods to discover people in microblogging environments. One method is based on search by specific information about people or search by specific keywords they use [65][83]. It is based on keyword search and limited to information explicitly declared by the users such as name, location, marital status, interests etc. Since the users often do not declare their interest areas and other information which may help finding them, it is usually a time consuming task to locate those who are of interest. The other method is based on ranking people by their popularity to suggest a list of popular people for a given interest area [38][21][89]. These people suggestion tools rank people by their popularity measured by the number of other people following them. However, popularity based methods make the popular people more popular while keeping the valuable but less popular ones hidden. In addition, popularity based ranking methods make it impossible to find correct people when the subject matter is to interact, communicate or simply ask questions to retrieve specific information.

Google has adapted its search engine to include the tweets in the search results. It uses the number of followers to rank the results. Hence, the popularity of the users is a search criterion. To avoid spamming it also considers ranking the tweets from users who are in the followers list of the user who sends search queries to Google. While doing this, Google aims to find the real time and the most up-to-date tweets which include the searched keyword from popular users. In our work, we focus on users, the overall contributions of them to find their areas of interests and discover similar people who are also interested in these areas.

There are numerous studies in the area of collaborative tagging as well. Some of them focus on discovery of information resources while some others focus on discovering users and communities. Here by, we briefly describe these works related to our thesis and explain how our research differs from them.

Tom Gruber, proposes a model for the tagging activity to structure the relations between users, tags and resources [4]. The main idea is to create a “collective intelligence” or “wisdom of crowds” out of the collaborative tagging systems. In his model he suggests that describing the tagging activity in a structured way would be aligned with the current efforts in the area of Semantic Web [34]. This model has been extended by many research to model tagging activity and collaborative tagging [12][25][26]. The TagOntology is about identifying and formalizing a conceptualization of the activity of tagging, and building technology that commits to the ontology at the semantic level [4]. Based the idea to create tag ontologies to discover resources and users semantically, the proposed models for tagging has been used.

Mika on the other hand proposes another model to discover semantic relations between tags by adding the social dimensions [23]. In this model he uses co-occurrence methods for tags, users and resources by defining a tripartite model of ontologies [23].

Some other research focus on extracting relations between tags based on semantic similarity measures such as cosine similarity, keyword co-occurrence and FolkRank [35][36]. Cattuto, analyzes the three of these methods in Collaborative Tagging Systems and compare them on a large-scale dataset from social bookmarking site del.icio.us [35]. His research shows that co-occurrence relatedness of the tags is suitable for discovering the concept hierarchies while cosine similarity and FolkRank are better at synonyms and multi word phrases respectively.

The research on word associations based on the co-occurrences of words has shown that the co-occurrence can help finding search terms for information resources [17]. Research driven by DERI proposes an algorithm to suggest similar or related tags for the resources in the collaborative tagging systems [37]. Their algorithm uses co-occurrence of words to extract associations between tags. Motto, proposes a model to discover semantics behind tags by using the co-occurrence methods to discover relation types such as is-a, has-a relations between pairs of tags in a cluster.

In this thesis, we focus on the content of the tags instead of modeling them to create tag ontologies. Our work is distinguished from the research in the area of tag ontologies

definition. Instead we focus on the content of the tags and their annotations with the users in order to discover community relations based on the tags users share. Co-occurrence of tags in the same microblog inspires us to generate relations between tags and other words which are not used as tag. In addition, co-occurrence methods allow us extend the set of keywords we associate with a user while exploring their interest areas.

3. PROBLEM STATEMENT

Our aim in this research is to identify the interest areas of users who publish content in the microblogging social applications. Furthermore, having the interest area of the users, we aim to identify communities emerging around specific interest areas.

As we mentioned in Chapter 2, UGC allows users publish information, share content and communicate directly with each other or through an engaged content. When people want to retrieve information, they either search for the valuable content or they look for people who can provide valuable information to them. However, it is not always easy to find the relevant people in a specific interest area or domain in order to retrieve the valuable information.

Users of the social web applications have attributes such as identity, reputation, groups, relationships etc. [3]. Social Web Applications today require their users to enter a set of attributes in detail which may or may not be part of the attributes mentioned previously. By means of user profile entry interfaces users explicitly define themselves by selecting an interest area among a predefined set of interests or entering free format keywords. Most of the time, these attributes are not enough to find relevant people. Main reasons for the difficulty of finding the relevant people are as follows:

- People do not always provide all necessary information regarding their specialties, expertise or interest areas
- Due to the limited entry facilities such as character limits, people may not be able to define all information about them
- Content of user contributions may possible not aligned with their area of interests.
- The information given by people to identify themselves are limited to the attributes which are requested by the social application
- Social search engines uses keyword matching to find related people but this requires that users should identify themselves with all possible keywords in order to be found, furthermore, requires searchers to know relevant words to search for them.

- People connect to other people who they already know, so other people who may also be related in terms of interest areas or specialties are not accessible

In this thesis, we analyze content published by the users of the social web applications in order to extract the interest areas of the users. Instead of solely relying on information explicitly declared by users about their areas of interest, content they contribute is used. We assume that we can find relevant people in a specific interest area by processing the content they publish. We also note that our focus is on people who publish content actively and aligned with their areas of interests. Hence, if an expert person in the area of photography for example does not post anything about photography but only publish text content regarding his daily life, this person is not in our scope in terms of interest area of photography. Besides, the more content people publish about a specific context, the more they expose their interest area.

Social web sites such as Twitter, Flickr, Del.icio.us, DeviantArt, have limited facilities to find and suggest users. These facilities are based on number of content they provide or number of relationships they have in the environment. However, none of these facilities are based on analysis of the content that users provide.

Our focus is to find people who provide relevant and valuable information by publishing UGC. We use text content that people share in the microblogging systems to understand the areas of interests of users. We then move forward to find the relationships between users in terms of interest areas.

Current relationships defined in Twitter do not allow other people to understand the type of the relationship in terms of interest areas. Though we know which people are friends by looking at their friends list, we can not say in which area these people share content or which common interest makes them connected.

A way to search for people in microblogging systems is to search for tags they use. Tagging is a way to categorize and organize the content. It also allows people to search for other users who publish content that contains a given tag. However, this method is restricted to the knowledge of the person who run the query since the person should know

exactly which tags define the searched subject and the user they look for. Besides, people who publish knowledge that other people would like to retrieve may not use tags for the content they provide since there is no restriction to use the correct keywords or tags for the content.

In our study, we also focus on expanding the set of words used by the people in the content they publish (microblogs in our case) so that other people can find them even if they do not use exactly the same keywords used in the search queries. We identify relevant keywords in an area of interest by using keyword co-occurrence method.

As described above, we have three main objectives in this research. First, we analyze user generated text content published in microblogging systems in the form of microblog posts, so called Tweets in Twitter. By analyzing the content, we identify specific interest areas that users publish content about. Next, we analyze the content published by other users in order to find relations between users around a specific domain to identify if any communities can be extracted based on common interests. Finally, we aim to expand the keywords related to a specific interest area by using the keyword co-occurrence method.

In the following section a sample case is described where the problems mentioned above are pointed out.

3.1. Sample Case

Consider a scenario that a Twitter user, User_A, is an expert in the area of digital photography. User_A likes sharing information with other people via Twitter including upcoming events, conferences, significant academic papers, practical information, trends and new technologies in the area of digital photography. User_A is a reputable person in the area of digital photography that he has thousands of followers and a bit less from the number of followers he has, he has a few friends in his list to follow.

Another significant information regarding the identity of the User_A is that he loves nature a lot and participates to the outdoor activities such as climbing and biking. He also publish content about his outdoor activities and information related to the nature sports.

Some users, with whom the User_A has friend or follower relationship, have common interest of outdoor sports with the User_A while other users are connected due to their interest in the digital photography. Some users may be connected because they have both interests in common.

Consider that User_B is a user who is in the followers list of the User_A and interested in the digital photography as well. Another user, User_C, who is in the friends list of the User_A is also an expert in digital photography but he likes publishing content about birds only. Finally, User_D, who is another expert in the area of digital photography is in the friends list of the User_A and publishes valuable content about the digital photography just as User_A. In addition, User_D, User_C and User_B has no follower or friend relations. Following table summarizes the information regarding these three users and the information they give about them in their profile.

Table 3.1. Sample users

User	Publish Content About	Relationship with User_A	Profile Information
User_B	Digital Photography	Follower	“From Istanbul”
User_C	Birds	Friend	“Digital Photography”
User_D	Digital Photography	Friend	None

Since User_B is a follower of the User_A and we know that their common interest is digital photography, User_B would also be interested in the content that User_D publishes. On the other hand, though we know that User_C is an expert in the area of digital photography, User_B would not prefer following the content related with the birds.

There are alternative ways in Twitter which allow User_B to discover User_D or any other users who are connected to the User_A and interested in the digital photography. One

alternative is to have a look at the friends list of the User_A and check the profile information given by the user. However, in our case User_B would probably miss the User_D by only checking the profile information since User_D has provided no specific information regarding his interests. User_B would also start following User_C due to the declaration of his interest in Digital Photography. In the second case User_B would receive information about birds which makes it difficult to distinguish the valuable information among irrelevant content. Lists or specific groups which the users are a member of, can also provide a clue about the users' interests. However, the usage of lists is again optional and users do not necessarily be a member of any lists. Besides, they may be publishing content about digital photography but be a member of lists such as books, tv series, literature etc. which do not have any relations with the area of digital photography.

Another alternative is to use applications specifically designed for Twitter such as Twitter's Suggested Users List [38] or Wefollow [21] to search for users who are interested in a specific area. Twitter's Suggested Users List is a facility which ranks the twitter users based on the criteria such as popularity, number of updates, number of followers and a few subjective profile information which might be interesting to other people especially the new users who do not know who to follow. Suggested List do not allow users search for users specifically related to an interest area and limited to the top 100 users ranked by the algorithm. Wefollow, on the other hand, uses similar criteria that the Twitter's Suggested Users List algorithm uses, but has more facilities. Wefollow categorizes users under specific tags and rank users in each category by their popularity based on the criteria similar to the Twitter's Suggested Users List. People can search by tags and retrieve the list of the most popular users under a specific category.

Both Twitter's Suggested Users List and Wefollow allows users find the most popular users in Twitter. However, it does not provide easy access to the people who publish valuable content but not as popular as to be listed in these tools. Another concern about finding users via these tools is that people would prefer connecting other users with whom they can interact with. User_B in our scenario, would not be able to discover that User_D is a valuable person in terms of digital photography unless User_D is one of the most popular Twitter users listed either in the Wefollow or Twitter's Suggested Users List.

The third alternative which we introduce in this thesis is to develop an algorithm to process the content of all users who publish content and connect users automatically based on the relevancy of the content they publish. In our sample case, we would find out that the User_D is publishing content in the area of digital photography by processing his tweets to understand his interest area. In addition, User_C would never be connected to User_B as long as User_C publish content about birds only.

In our study, unlike associating the users with specific words such as digital or photography, we also associate the user and other users connected to the initial user with other possibly relevant keywords. For instance, User_A may be using the keyword digital_photography either as a word or hashtag #digital_photography frequently but never use the word photoshop throughout his tweets. Assume that another user, User_E, publishes information about photoshop. Once we extract that they are related in the context of digital photography area as a result of our study, we also provide connection between the words photoshop and digital photography. This makes it possible to extend the tag cloud of the users.

In the next chapter we explain our analysis on Twitter in order to understand the dynamics of the environment, then propose our method in detail in the following chapters.

4. ANALYSIS

In this chapter, we explain our experiments we performed to understand Twitter environment. We describe the method we used for each experiment, the data that we used as input and we explain our findings in detail. We start our experiments by gathering publicly available data from Twitter using its java API library Twitter4J [39]. The content we received is evaluated in terms of;

- tweet structures,
- users' behaviors in Twitter,
- relation between interest areas of users and the content they provide and
- relationships between users around a specific interest area.

Each analysis at this phase moved us to the further step during the development of our proposed model. Our findings at each step are also explained separately in each section below.

4.1. Understanding Twitter

Analyzing Twitter as a whole, covering all the data in it, requires allocation of too many resources and a completely different study which is out of our scope. Instead of trying to capture all the data from Twitter and analyze it, we selected a few samples to work on. In addition, we refer to the services, available in the Internet, providing the statistics on Twitter such as TweetStats[40] and TwitterFacts[41] while evaluating the results of our experiment. Besides, we refer to the social marketing specialists who also work on Twitter to understand the structure of the tweets, common behaviors of users and create models for the retweets [77].

We initiated our experiment to understand the dynamics of Twitter and understand the structure of the tweets by gathering the data which are publicly available to all Twitter users. When users log in to Twitter, it displays a set of keywords which are the most

popular ones used by Twitter users. A set of these popular keywords are selected from the main page and a search query is sent via Twitter API.

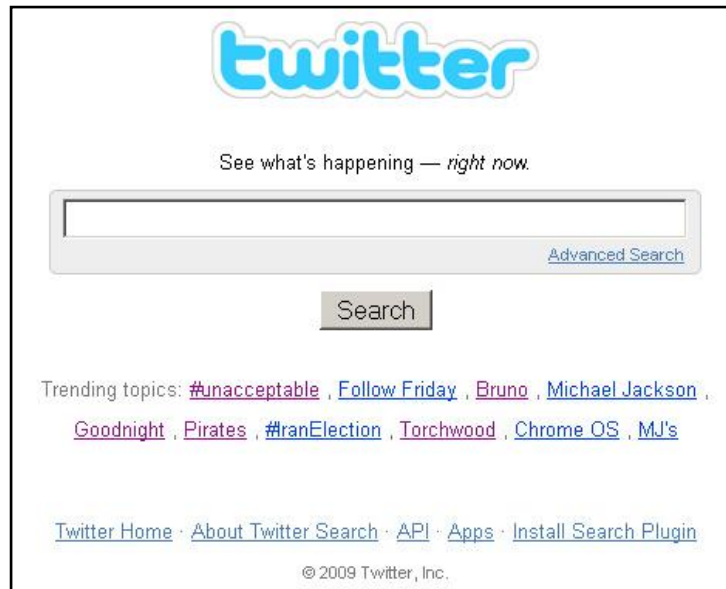


Figure 4.1. Twitter search [65]

Table 4.1. Data for the experiment – understanding Twitter

Query	# of Tweets	Time Period
#Swineflu or swineflu	250,507	29/4/2009 3/5/2009
#Wolverine or Wolverine	76,636	1/5/2009 3/5/2009
#Obama or Obama	91,554	11/4/2009 18/4/2009

As a result of this experiment, we noticed that people who use twitter for commercial activities such as advertising include the keywords which are listed in the trending topics list frequently. This behavior causes that tweets which do not contain any information about the topic is retrieved just because it includes the keyword.

We also see that the use of conversational phrases and stop words are frequently used. Another finding of our analysis is that links are used due to the character limitation of tweets in the form of tiny url. TinyURL[66] is a web service that provides short aliases for redirection of long URLs. People use links very frequently when they want to share their knowledge or give information about a specific subject.

Table 4.2. Swine flu tweets

Swine Flu tweets
Check out : Fighting Swine Flu http://tinyurl.com/dfl6bp
Virus keeps spreading around the world - The toll from the swine flu epidemic appears to be stabilising in Mexico, but officials s http: ...
swine flu scares me because im not really sure how one contracts it
LOL @ Spedi going to Mexico for their honeymoon. Please GOD, let them get infected with swine flu.
This train commute ain't gonna be funny if this swine flu takes off. Some people are saying it may kick in hard this winter. Great...

Retweets are the shared content published by a user and shared by other users containing the reference to the original user who published it before. We also see that more than ten percent of the content we retrieved contains retweets. This shows that not every user creates content but they also help spreading the information by using the retweet functionality Twitter.

Below are the statistics we gathered as a result of our analysis. It is shown in the table that at least one of the ten tweets either contain a link or a retweet. Besides, the ratio of the links and retweets usage change depending on the subject. While reviews and comments are shared as it is the case for the movie “Wolverine”, the number of retweets or links decreases. The link usage increases when the subject matter of the tweets is more informative such as news about “Obama”.

Table 4.3. Results for the experiment – understanding Twitter

Query	# of distinct users	#of links/ tweets	# of retweets/ tweets
Swineflu	163,330	80,984/250,507 = %23	44,904/250,507 =%17
Wolverine	55,563	8,526/ 76,636 = %11	7,506/ 76,636 = %10
Obama	43,069	50,935/91,554 = %56	27,229/91,554 = %29

In our research we focus on the content which is in the form of microblogs named tweets. The results of this initial experiment show us that we should refine the content of tweets as to work solely on the keywords which possibly belong to an area of interest. In other words, we should eliminate the words which are irrelevant to any specific interest area.

Another observation in this experiment is that hashtags, tags with the hash sign, are also used together with the links and informative content. But since we inquired tweets by hashtags such as #swineflu, #wolverine and #obama and most of the tweets already contain these hashtags in this experiment, it does not show any statistics regarding the usage of hashtags among all tweets. We analyze further the usage of tweets in the following experiments.

4.2. Understanding User Behaviors

As we explained in the Chapter 2, our main assumption in this thesis is that the content that the users publish is aligned with their interest areas. In order to see if the users behave in parallel to our assumption, we performed this experiment. In addition, we aim to see that current search facility of Twitter is not enough to find relevant people of a specific interest area.

We selected the interest area of football and picked up a few keywords which were popular during the time period that we collected data. Our search inquiry to retrieve data included the keywords Barca, Barcelona, Man-U and Manchester United. The data is collected around the days before and after the Champions League games when they were the trending topics. After we collected the data containing the keywords, we ranked users by the number of tweets they publish containing at least one of these keywords.

Table 4.4. Data for the experiment – understanding user behaviors

Query	# of tweets	# of words	# of distinct users	Time Period
Barca, Barcelona, Man-U, Manchester United	54,085	837,583	35,570	12/5/2009 31/5/2009 (not continuously)

Table 4.5. Results for the experiment – understanding user behaviors

user_name	tweet_count
hotel_barcelona	736
Soccer_Wire	458
Book_Manchester	155
ManUtdNewshound	152
Sportsfirst	121
PiperQ	113
TelegraphMG	100
Trading_System	99
Strodnews	92
BetOnFinal	89

As a result of this experiment we see that the user “hotel_barcelona” is on top. However, it is not a human user but an application that publish content periodically. Besides, the content it publishes is not related with football. Other users’ tweets are also automatic updates from applications which use Twitter API.

In our thesis, we use keyword co-occurrence methods in order to eliminate irrelevant content which may be contained in the data due to the keyword matching functions. This method also shows us that relying only on the number of tweets which contains the search query keywords is misleading since we focus on human users instead of applications automatically publish content which are called bots.

This experiment also shows us that it is possible to avoid bots by using the relationships between users based on conversations. What we mean by conversations in Twitter is mention, reply and direct messaging relationships as they are explained in Chapter 2 in detail. In our model, we consider the users who have reply relationship while exploring the interest areas.

The following experiment was performed to see if our assumption regarding the conversational relationships and users' relevancy based on the specific interest areas are correct or not.

4.3. Understanding Relations between Users of a Community

This experiment is needed to see if the users who have conversations among them are candidate members of a community of a specific area.

Similar to the method we implemented in the previous experiment, we selected a keyword which we know that belongs to the specific interest area which is football in this case. Hashtag #Arsenal, which was on top of the popular trends list of Twitter at the time we initiated this experiment, was selected. All the tweets that were publicly available and contained either the hashtag #Arsenal or the word Arsenal were retrieved. Among these tweets, the most mentioned or replied or retweeted users were selected. After extracting these users, we retrieved their tweets as well. This experiment was performed with a very limited set of data which was collected one day before and after the day of the football game between Arsenal-Barcelona.

Table 4.6. Data for the experiment - understanding relations between users of a community

Query	# of Tweets	# of distinct Users	Time Period
Arsenal	91	34	12/5/2009 14/5/2009

After we retrieve all the data, we analyzed the words in them in order to see the most frequent words and their relevancy to the football domain. The results of this experiment are shown below:

Table 4.7. Results for the experiment - understanding relations between users of a community

Tag	Count
Arsenal	31
Barcelona	11
Champions	2
Disappointed	2
Fan	2
Football	5
Messi	5
Player	3
Soccer	4
Spain	2
Twitter	2
World	2

As shown in Table 4.7, the word Arsenal is on top due that it was included in our search query. We see that other words in the list are all relevant words to the area of football. Since the data was collected during the time of champion's league games, we also see that the words are specifically related with the game. This results show us that more relevant content relationships can be extracted by using the conversational relationships between people. Besides, we also see that tags, which are used by a set of users who share common interests, are similar.

In our proposed model, we want to focus on human users instead of bots and spammers. As a result of this experiment we see that more relevant content can be extracted by analyzing the tweets of users who have conversations. We also get help of this experiment while collecting our test data which help us avoid bots and spammers.

Our experiments so far were performed from a perspective of content. We gathered publicly available data from Twitter during a period of time and analyze the content in order to see the structure of the tweets. We also try to understand user behaviors and usage of tweets from a perspective of a set of users who share common interest. In the next experiment, we try to understand the content published by a specific user. Instead of retrieving the users from the content, we retrieve content published by a user.

4.4. Understanding the Tweets from Single User Perspective

Our aim in this experiment is to analyze the content published by a single user. We assume that the content published by a single user reflects the interest area of that user. The more content they publish about their interest area, the more they expose themselves and easier for us to discover such users. In order to see if our assumption is correct, we picked users who are listed in the most popular users list of Wefollow[21] based on specific categories. Afterwards, we retrieved all the tweets published by these users. Time period is not a parameter in this case since we retrieve all tweets published in the past until the time we inquired the user.

Table 4.8. Data for the experiment - understanding from single user perspective

User	Category	# of Tweets	# of distinct words/ all words	After Pre-Processing # of distinct words/ all words	# of tags
Steve Simon	Photography	972	3,301/12,564	2,712/7,008	12

Table 4.9. Results for the experiment – tags of a user

Tags	Count
'photographer'	3
'photography'	3
'4k'	1
'23'	1
'journalist'	1
'photoj'	1
'photo'	1
'nikon'	1
'1'	1

Table 4.10. Results for the experiment – words of a user

Words	Frequency
'ss'	71
'nikon'	38
'photo'	28
'workshop'	25
'looking'	24
'aperture'	23
'work'	18
'post'	17
'photography'	16
'guy'	16
'flickr'	15
'camera'	14
'video'	14
'stuff'	14
'world'	14

We analyzed the words and tags of the users and ranked the most frequent words and tags used by them. In Table 4.9 tags used by the user “stevesimon” is listed. In Table 4.10 As a result of this experiment, we see that tags are used rarely but when used they give keywords regarding the interest area of the users. In addition, we also see that the most frequent used words and tags are related to each other. This also supports our assumption that we can extract the interest area of a user from the content they publish.

4.5. Summary

Our experiments are explained in the preceding sections of this chapter. In addition, brief description of how we apply our findings we get out of our analysis is given at the end of each section. Hereby, we summarize the results of our analysis as follows;

- keywords which are listed in the trending topics list are subject to advertising or spamming
- stop words such as to, at, in, are, just etc. and conversational words such as "I think", "good morning" etc. are used frequently
- links are frequently used
- retweets are frequently used
- bots are on top of the list when we inquire by specific keywords; especially the words listed in Twitter's Trending Toppics
- users who have conversations are likely to be human users.
- words and tags, which are used by a set of users who share common interests, are similar
- hashtags are used rarely, but when used they give keywords regarding the interest area of the users.
- the most frequent words and tags, used by a single person, are related to each other

In the following chapter, we propose our model which is shaped by the result of our experiments. Our model to explore specific interest areas and connected users around these interest areas are explained in detail in the next chapter.

5. PROPOSED MODEL

In this chapter, we propose a method for identifying interest areas of users and the communities based on these interests.

Microblogging allows users to expose their ideas, feelings, interests, knowledge and expertise by means of short text messages. A microblog is a collection of these short messages each of which is called a microblog post. Users also interact through microblog posts. To express desired content in a space efficient manner, various space conserving conventions and notations have emerged. For example in Twitter, hashtags are tokens that start with a hash symbol (#) prefix. They are used to tag microblog posts (tweets) they occur in. When collectively used, they group, organize and filter related microblogs.

This thesis focuses on identifying the context of microblogger contributions. Context is to be interpreted as an area of interests of microbloggers. For example, the context of contributions from a microblogger who is interested in bird watching is expected to have keywords related to the bird watching interest area. Note that such context may not always exist and that we are interested in discovering those that exist. Furthermore, individual microbloggers may have similar context that they can build up a community of common interests.

This thesis proposes to:

- (i) identify microbloggers interests given their contributions
- (ii) identify a community of interest given a microblogger
- (iii) examine social network properties of microblogger communities of interest

In order to achieve these, first the nature of a microblogger's contributions must be identified. This is not as trivial as one might imagine since a microblogger's contributions:

- may relate to numerous topics
- are fragmented thoughts into many microblog posts
- may be a duplication of another microblogger's contribution
- may contain references to other users, external links, tags etc

- consist of phrases and cryptic contributions motivated by spatial and temporal constraints due to the character limit.

In short, microblog posts do not consist of well formed sentences, much less thoughts. In contrast, they are similar to streams of consciousness where many different fragments of thought may surface in a non-linear fashion.

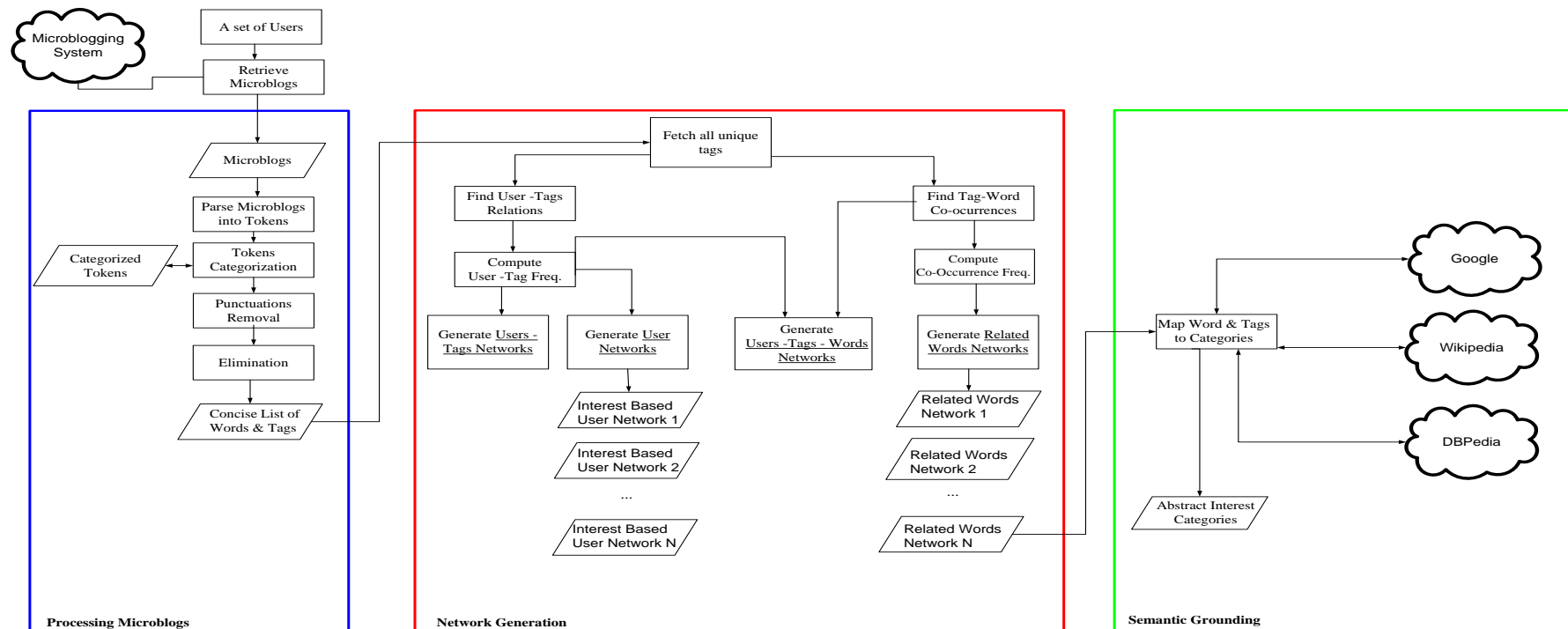
Our approach is basically to first process a collection of microblog contributions from individual microbloggers and reduce them into a set of keywords to represent the nature of their content. Secondly, to identify a community of interest based on the common context posted by different microbloggers. Microbloggers who use the same tags in their microblogs are considered to post common context. The properties of communities are further used to describe its members.

The resulting community of interest serves those who wish to observe and/or participate in contributions in an area of interest within a microblogging platform. In another words, the intent is not to simply discover people, but observe or take part in an evolving expression platform. The aim here is to provide microbloggers better support for locating contributor networks relevant to their interests.

Our method in general consists of two steps shown in Figure 5.1. :

- (i) Processing Microblogs
- (ii) Networks Analysis

Semantic Grounding step is shown as well in the proposed model. Based on our initial experiments after the implementation of this step, we decided to leave it as a future work.



Abstract Interest Categories: A category under which all the nodes in each related words network can be gathered

Related Words Network: Network of words and tags where words are connected to tags based on co-occurrence relations and all connections are weighted by the frequency of their co-occurrence.

Users - Tags - Words Network: Network of words, tags and users where users are connected to the tags they use and words are connected to the tags they co-occur with. This network emerges by combining the related words network and the user-tag network.

Token Categories: Stop Words, Links, Tags, Emotion Words, General Words, Time Related Words, Mentions & Replies

Users -Tags Network: Network of users and tags where users are connected to the tags they use.

Interest Based Users Network: Network of users where users are connected to each other if they use common tags. The connection is weighted by the number of common tags they use.

Figure 5.1. Proposed model

As shown in the Figure 5.1, user contributions are processed in the following manner:

- A user set is selected, such as a set of:
 - (i) randomly selected users
 - (ii) users in a list constructed by themselves or third parties
 - (iii) users who use a search keyword such as “obama”, “swineflu”, “semanticweb” in their posts
 - (iv) users who post between a given time interval such as March 1st, 2009 – March 31st, 2009.
 - (v) users known to interact with each other
- All microblogs published by selected users are retrieved
- Microblog posts are parsed into tokens to have a raw set of tokens
- Tokens are categorized into the categories (Section 5.1.1)
- Insignificant tokens are filtered out (Section 5.1.3) to have a concise list of tags and words
- Area of interests, namely related words network, is extracted
- User-tag networks are generated to associate users to the areas of interest
- Users – tags - words network is generated by combining related words and user-tag networks in a single view
- User networks are constructed based on common tag usage

The aim here is to determine area of interests and microbloggers who contribute and interact based on those interests. To achieve these aims, two types of relations are defined. Relations between tags and words are used to determine area of interests. Relations between tags and users are used to determine interest areas of users, additionally, communities of microbloggers who contribute and interact based on those interests. Tags are considered as essential part of this model to identify both types of these relations.

The relations between tags and words are used to identify areas of interests. Area of interest is defined as a set of related words which are associated by weighted co-occurrence relations. Words are introduced by microblogger contributions. Tags are intentionally provided to categorize and later retrieve microblogs. Since tags categorize and annotate

microblogs and are deliberately provided by their contributors, we can consider them as highly significant. It is likely that they are strongly associated with other words and tags they co-occur within a post. Furthermore, it is also reasonable to assume that microbloggers who use the same tags are related. Since tags are freely chosen by users it is possible that tags are polysemious. However, taken in the context of co-occurrence with similar words and users it is generally possible to disambiguate.

Keyword co-occurrence is a semantic similarity measure, which is applied in collaborative systems to discover semantics of tags [37][17]. There are other methods such as FolkRank and cosine similarity to apply to collaborative systems in order to discover semantics of tags. The methods keyword co-occurrence, FolkRank and cosine similarity are compared to understand which measure is better at finding semantics of tags in collaborative systems, and shown that co-occurrence similarity measure is suitable for discovering the concept hierarchies while FolkRank is better at finding multiword lexemes and cosine similarity is better at finding synonyms [17]. In this work, the focus is to find words which have different meanings but are related to an area of interest. For example, we focus on discovering relations between words such as opensource, linux, software, programming tools, free instead of relations between words such as opensource, open-source, open_source,oss etc. Therefore, we consider that extracting relations between concept hierarchies, in other words, applying keyword co-occurrence method best serves to categorize words in areas of interests.

Three steps of the algorithm proposed in this model are explained further in the following sections of this chapter.

5.1. Processing Microblogs

Microblog posts, due to their social nature contain many words and phrases which are not specific to any given interest, such as ‘I think’, ‘thanks’, ‘great’ etc. Also, most punctuation marks are of no interest.

Thus, the set of microblogs must be distilled to a set of words consisting of relevant keywords. The idea here is to determine a set of words that describes a microblogger’s

interest. The aim is to distill all the words uttered by a microblogger into a concise set of relevant words. To achieve this first, tokens are categorized, then tokens within insignificant categories are filtered out as is described in the forthcoming sections. The processing algorithm is shown in Figure 5.2.:

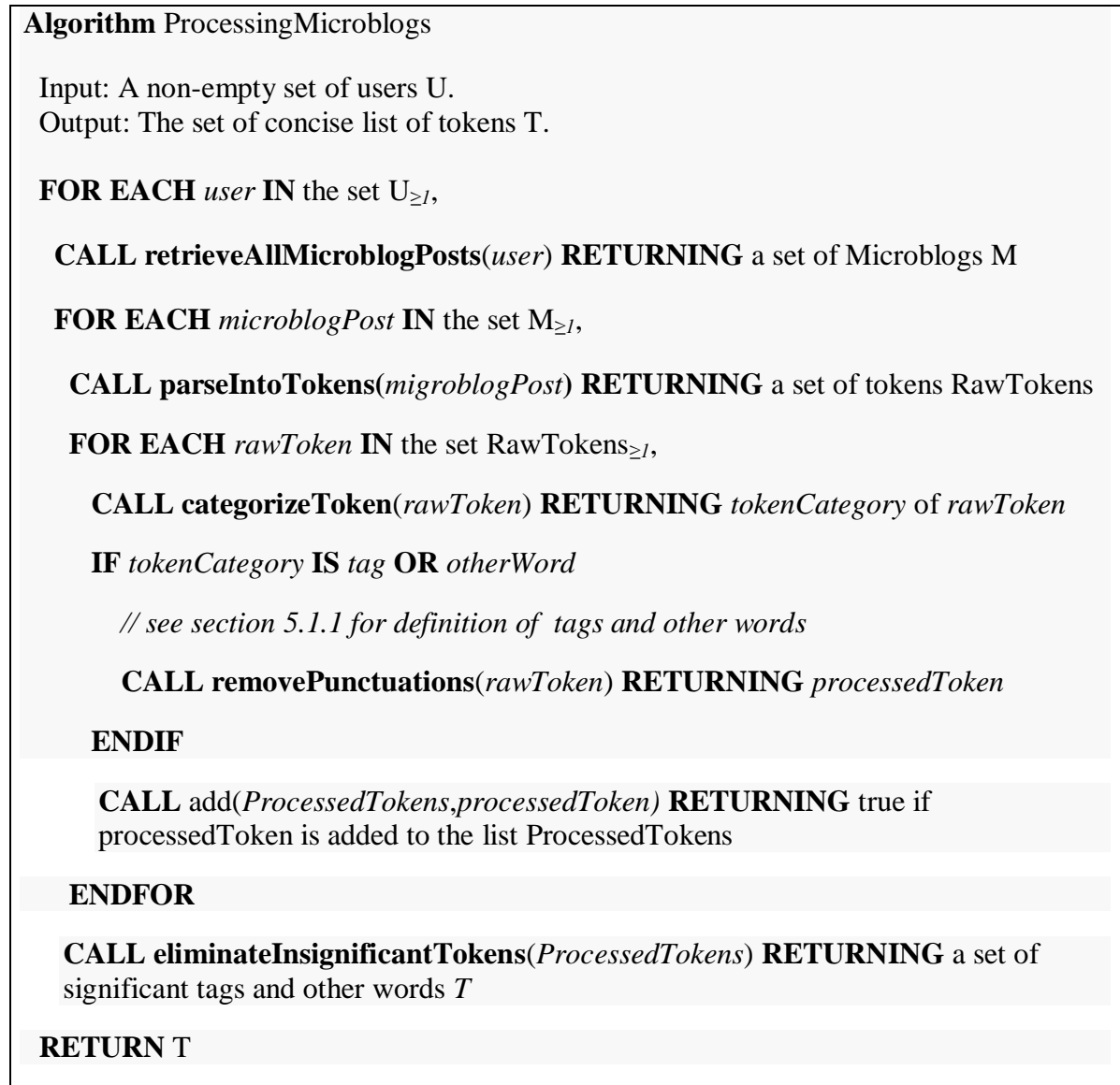


Figure 5.2. Processing microblogs algorithm

All microblogging contributions are first parsed resulting in a set of tokens. This set is referred as the set of raw tokens throughout this study.

5.1.1. Tokens Categorization

Tokens in the raw set are categorized into the various categories for further processing. The categories we consider are:

- **Stop Words:** Stop words are words that are either insignificant such as prepositions, articles etc. or so common that they can not be related to any specific interest area [18]. They vary from language to language and system to system. We consider English stop words only. See Appendix A for our stop words. All stop words are removed from the raw tokens set.
- **Links:** Microblogs are short messages. The real message is often found in an external link, which is provided in the post. All tokens starting with the character set “http” are categorized as links. All links are removed from the raw tokens set.
- **Tags:** Tags are used to associate a keyword with a post. They are similar to typical tags except they are part of the post. Thus, contributed by the poster and not by a third party. When other microbloggers adopt the same tag use, it results in grouping associated microblog posts. Figure 5.3 demonstrates three microbloggers who use common tags in their microblog posts. Since tags are meant to provide meta information and they occupy valuable space, they are considered as highly significant in identifying user interests.

Special notations are used to differentiate tags within a post. In Twitter, tag is a token with a hash (#) sign prefix. Due to the character limitation of microblog posts, hashtags are used to allow users to give brief information about the content of the links, pictures or simply the text they publish. Hashtags also used to search for what users had posted previously. Throughout this study, the terms tag and hashtag are used interchangeably.

In this work, tags are used as a mechanism for collective filtering. Tags sometimes correspond to know words and other times they do not. If a tag corresponds to a word we consider that word more valuable than if it occurred simple

as a word. For example, “#computer” is more significant than “computer”. In most cases, tags are abbreviations or set of characters that can not be found in a dictionary. Some examples of such tags are “#sle09” which is a name of an academic conference, “#oop” which is an abbreviation for object oriented programming. Sometimes a tag, such as “#clockout”, “#followfriday”, “#mm” or “musicmonday”, is used by many users having a special meaning which can only be known by a community who uses it.

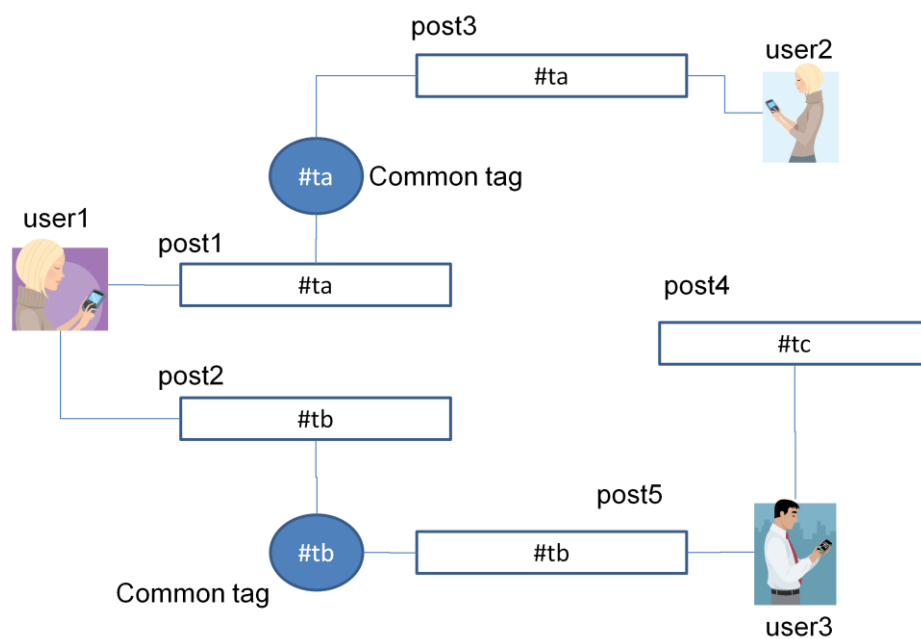


Figure 5.3. Common tag usage by different microbloggers

- **Mentions & Replies:** Posts include references to other microbloggers. We extract such users and classify them as replies or mentions. In general, user names with at sign (@) prefix are used in microblogs to reply to or mention a microblogger. In this study, all tokens with at sign prefix are considered as user names. Replies and mentions are differentiated by using their position in the microblog. It is assumed that a token is a reply if it has at sign prefix and in the first position in microblog. A token is considered as mention if it is not in the first order in microblog. Not all tokens with the at sign (@) are usernames and position of the token does not always identify that it is really a mention or reply. However, when general usage in social

networks and the nature of microblogging is considered, it is expected to categorize most of the user names by applying this method.

Example:

Microblogger1: @delta_goodrem Hope u had a Beautiful xmas Delta!

In this example, “@delta_goodrem” refers to a microblogger who is replied to in this post.

Microblogger2: Leaving house for first time since Xmas eve. Excuse to wear red duffle coat from sister and sparkly scarf from @evwa!

In this example, “@evwa” refers to a microblogger who is mentioned in this microblog.

All mentions and replies are removed from the raw tokens set since they are not significant words in terms of interest area specification.

- **Common Words:** Some words are used very frequently and are common to many posts. Examining microblogger behavior demonstrates common patterns of expression. The specific language used on these platforms introduces additional stop words. For example, we observe that social media related terms, temporal expressions as well as the aspects of the platform itself are used very frequently. Examples of such words are “free”, “check” etc. They are not effective in discriminating user interests. Therefore, such tokens are categorized as common words and eliminated from the concise list of words and tags. The selected words in this category are:

$$\text{Common Words} = \{ \text{“twitter”, “RT”, “RT@”, “tweet”, “free”, “check”, “good”, “big”} \} \quad (5.1)$$

- **Emotional Words:** When people post microblogs, they express their feelings as well. Tokens which express feelings of users are categorized as emotional words. Emotional words are also eliminated in our model.

$$\text{Emotional Words} = \{\text{"awesome"}, \text{"amazing"}, \text{"fine"}, \text{"nice"}, \text{"bad"}, \text{"beautiful"}, \text{"love"}, \text{"enjoy"}, \text{"hate"}\} \quad (5.2)$$

- **Time Related Words:** Tokens describing the time of the events or information is frequently expressed in microblogs. This is because people write frequently (several times a day in general). Microblogs are very much about “now”. Past is very recent and future is very near future.

Though there are many time related words in English language, the most common ones are defined as time related words as follows in our model:

$$T = \{\text{"today"}, \text{"tomorrow"}, \text{"now"}, \text{"2009"}\} \quad (5.3)$$

All time related tokens are removed from the raw tokens set since they are not specifically related to any interest area.

- **Other Words:** In the set of raw tokens, the tokens, which are not categorized under any of the link, mention, reply, common word, time related word, emotional word and stop word categories fall into other words category.

The set of all tokens in the tags and other words categories are named as concise list of tags and words. The concise list of tags and words are considered as candidate keywords to determine areas of interests after the punctuation marks are removed from these tokens if exist.

Next section describes the punctuations removal process in detail.

5.1.2. Punctuation Removal

Punctuation marks are non-alphanumeric symbols used in written language. Alphanumeric characters are defined as the set of numbers 0 to 9 and letters A to Z. All characters which are not in the alphanumeric characters set are considered as punctuation marks in this work.

Punctuation removal is the process of removing all punctuation marks from the tokens in the concise list of words and tags (i.e. “favorite!!” becomes “favorite”). In microblogging environments punctuation marks are frequently used as attached to the words. Punctuations are removed from the tokens in the concise list of tags and words in order to avoid interpreting the tokens “favorite!!” and “favorite” as if they have different meanings.

Punctuation removal is done after the categorization of the tokens, since categorization depends on special tokens which use punctuation marks (i.e. hashtags, mentions, links etc.).

5.1.3. Elimination

The last step of processing microblogs aims to filter out tokens insignificant to area of interests. The elimination process removes links, stop words, emotional words, common words, time related words, mentions and replies from the raw tokens set. Only tags and tokens categorized under other words are not eliminated. The resulting set is called as the concise list of tags and words. This concise list is analyzed to reveal user interest areas and associations to other users.

5.2. Networks Analysis

The interest based social network formation consists of three parts; determining a set of words relevant to an interest area, determining the interest areas of users and identifying interest based communities.

Each step mentioned above is modeled as a network of nodes and edges at an abstract level. The first network is named as “Related Words” network and represents a set of words relevant to an interest area. The second one is named as “Users – Tags” network and represents the associations between users and their interest areas. The third network is named as “Users – Tags – Words” network which is a combination of the two previous networks and represents associations between users, their interest areas and words relevant to interests areas in a single view. Finally, “Interest Based Users Network” is extracted by

using the common interest areas defined in “Users – Tags” network. These network models are explained further in the following sections.

A set of microblog posts posted by a set of given users is defined as P . Our model is not dependent on the selection of the user set. Any set of users as described in Section 4, can be selected. Therefore, we do not restrict the definition of P based on the selection of users.

$$P = \{ p \mid p \text{ is a post} \} \quad (5.4)$$

A microblog post consists of a set of tokens which can be in a category such as tag, other words, links, mentions etc. These categories are defined in Section 5.1.1.

A formal definition of a post can be defined as:

$$\text{post} = (\text{stop word} \mid \text{mention} \mid \text{reply} \mid \text{link} \mid \text{common word} \mid \text{time related word} \mid \text{emotional word} \mid \text{tag} \mid \text{other word} \mid \text{punctuation marks})^* \quad (5.5)$$

Three types of nodes are defined to model mentioned networks:

$$T = \bigcup_{i=1}^n t_i \quad (5.6)$$

$$W = \bigcup_{i=1}^m w_i \quad (5.7)$$

$$U = \bigcup_{i=1}^k u_i \quad (5.8)$$

- T : A set of tokens categorized as tags in Section 5.1.1.
- W : A set of tokens categorized as other words in Section 5.1.1.
- U : A set of given users who use at least one tag.

These node definitions will be referred while defining the network models in the following sections.

5.2.1. Related Words Network

This network represents the set of interest areas of a given set of users. An interest area is defined as a set of related words associated based on their co-occurrence. All tokens from a given set of users are processed resulting in a concise list of words and tags which represents the nodes in the Related Words network. If a token in the concise list co-occurs with another token from the same list, an edge is formed between these nodes, i.e. tokens. . Co-occurrence frequency of these nodes gives the weight of this edge. An example is shown in the Figure 5.4.

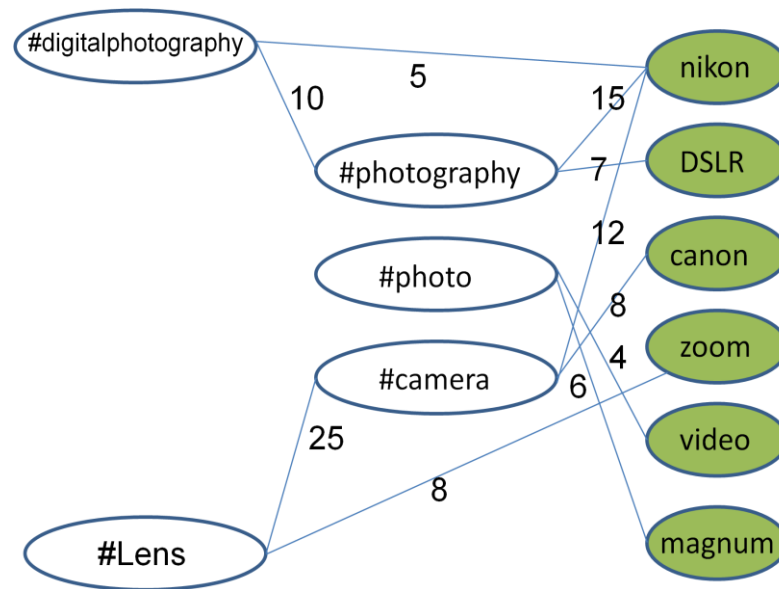


Figure 5.4. Sample Related Words Network

Tags are key elements in this network. The nature of microblog posts brings up the problem of handling too many words which are insignificant to an area of interest (i.e stopwords, usernames). Constructing edges between tags and other words using co-occurrence frequency metric, is considered as a better approach than constructing edges between all the nodes in the union set of T and W. Due that tags are meta data about the content embedded in the microblog posts, we consider that tags give more specific information about the context and the areas of interests than other words. Therefore, it is aimed to filter out most of relations between less relevant words which are categorized under other words set.

Based on the approach explained above, the following steps are executed to generate the related words network. The algorithm that performs these steps is given in the Figure 5.5.

- Step1: Retrieve all the words used as hashtag at least once
- Step2: Find all the words co-occur with the words found in the first step at least once
- Step3: Find frequencies of each co-occurrence
- Step4: Generate related words network

```

Algorithm GenerateRelatedWordsNetwork

Input: A non-empty set of categorized tokens T.
Output: The network of related words N.

CALL retrieveAllDistinctTags(T) RETURNING DistinctTagsList

FOR EACH distinctTag IN the set DistinctTagsList≥1,

    CALL retrieveCooccurringWordsWith(distinctTag) RETURNING
    CoOccurringWordsList

    FOR EACH CoOccurringWord IN the set CoOccurringWordsList≥1,

        CALL createEdgeBetween(distinctTag,CoOccurringWord) RETURNING an
        edge

        CALL calculateCooccurrenceFrequency(distinctTag,CoOccurringWord)
        RETURNING the weight

        CALL
        addNodeToRelatedWordsNetwork(distinctTag,CoOccurringWord,edge,weight)
        RETURNING a set of nodes N

    ENDFOR

ENDFOR

RETURN N

```

Figure 5.5. Generating Related Words Network algorithm

Depending on the microblog posts of the given set of users, a Related Words network may consist of one or more areas of interests. In the Figure 5.6, a sample Related Words network is given where connected and disconnected nodes are shown. There are also strongly connected and loosely connected nodes. An area of interest can be defined by setting thresholds for the node degrees to group the nodes which are strongly connected.

Similarly, clusters of nodes can be defined to identify areas of interest in a Related Words network. In this work, the focus is to find user relations based on common interests. Therefore, areas of interest are considered as related words, which are associated in the Related Words network and do not go further to define thresholds due to focus mainly on user relations.

A sample words network is given below for the words and tags which co-occur at least 8 times in the same microblog post:

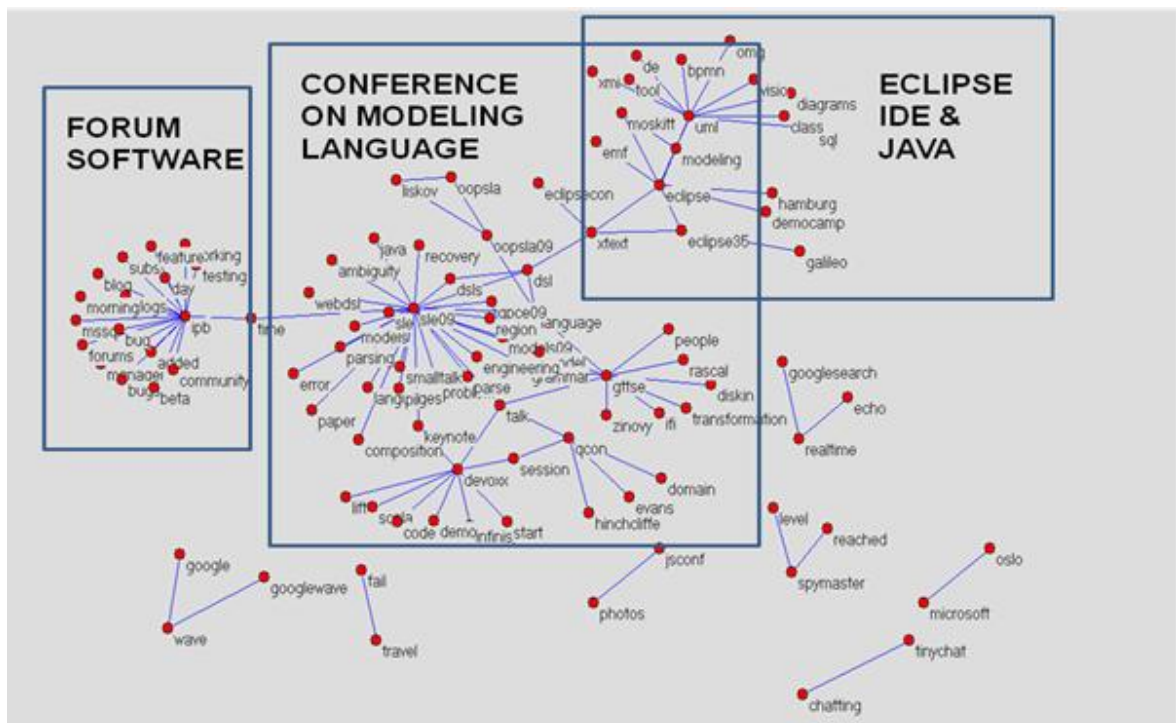


Figure 5.6. Sample Related Words Network

Related Words network consists of the union of the tokens in T and W. It is possible that a word which is used as a tag in one microblog post, can also be used as a word in another microblog post. In this case, the token will be an element of both the set T and W. For example, #photography is an element of T, but it is also an element of W since it is used as photography.

A common behavior in microblogging, is to use both word and tag form of the same token in a single post. For example, “jack0217: photography to magic and childhood

[#photography](http://bit.ly/cvoP5U)". Based on the co-occurrence relations of words in W and tags in T proposed in this work, photography and #photography should be connected. However, the focus in this work is to discover relations between different words. Therefore, self reference is not allowed in Related Words network. Defining the nodes as the union set of T and W allow us avoid self reference.

A Related Words network is defined as a graph with a set of nodes and undirected edges.

$$R = (V_{tw}, E_{tw}) \quad (5.9)$$

The set of all nodes in the Related Words network of a given set of users is represented as V . A node $v \in V_{tw}$ is defined as the element of the union set of T and W .

$$V_{tw} = \{v_{tw} \mid v_{tw} \in T \cup W\} \quad (5.10)$$

If a tag $t \in T$ co-occurs with a word $w \in W$ in the same microblog post $p \in P$, there is an edge between t and w . The set of edges in a Related Words network is $E_{tw} \subseteq T \times W$ and can be defined as:

$$E_{tw} = \{(t, w) \mid (t \in T) \wedge (w \in W) \wedge (t \neq w) \wedge (\exists p \in P: \text{co-occurs}(t, w, p)) \wedge (\exists \text{weight} \in \mathbb{N}^*: \text{co-occursFreq}(t, w))\} \quad (5.11)$$

where $\text{co-occurs}(t, w, p)$ returns true if a tag $t \in T$ and $w \in W$ co-occurs in a post $p \in P$ and $\text{co-occursFreq}(t, w)$ is the total frequency of a tag $t \in T$ and $w \in W$ co-occurring in all microblog posts.

Weights can be used for two purposes. First, it can be used to filter out less relevant words in an area of interest. For example, by setting a threshold to remove the edges with a weight under this threshold would result in a set of connected nodes (tags and other words) which are more relevant in terms of co-occurrence relatedness to an area of interest. Secondly, in the case that more than one area of interest exists in a Related Words network, is to cluster related words belonging to different areas of interest. Defining thresholds

would be in place to implement the second one as well in order to find groups of related words. In this model, the weights are calculated but not used for any of these purposes due to the mainly focus on user relations.

Once the interests are extracted, the next step is to associate these interest areas with the users.

In the following section, we describe how the users and their interest areas are associated in detail.

5.2.2. Users – Tags Network

This network relates users to their interest areas. Users are associated with the tags they use. The association of a user and a tag is weighted by the number of microblog posts that a tag occurs in those posted by the user.

A sample representation of this type of network is shown in the Figure 5.7.

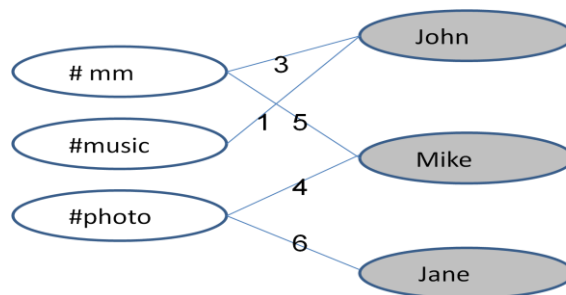


Figure 5.7. Sample Users-Tags Network

Tags are used to determine interest areas of users. Users who do not use any tags in their microblogs cannot be associated with any specific interest area with this model.

A Users - Tags network is defined as a graph with a set of nodes and undirected edges.

$$G = (V_{ut}, E_{ut}) \quad (5.12)$$

The set of all nodes in the Users - Tags network of a given set of users is represented as V_{ut} . A node $v \in V_{ut}$ is defined as the element of the union set of T and U .

$$V_{ut} = \{v_{ut} \mid v_{ut} \in (T \cup U)\} \quad (5.13)$$

If a user $u \in U$ posts a microblog containing a tag $t \in T$, there is an edge between the tag t and the user u . The set of all edges in a Users-Tags network is represented as $E_{ut} \subseteq (U \times T)$ and can be defined as:

$$E_{ut} = \{(u, t) \mid (u \in U) \wedge (t \in T) \wedge (\exists p \in P: \text{uses}(u, t, p)) \wedge (\exists \text{weight} \in N^*: \text{weight} = \text{usesFreq}(u, t))\} \quad (5.14)$$

where $\text{usesFreq}(u, t)$ is the number of posts in P where a user $u \in U$ uses a tag $t \in T$ in a post $p \in P$. $\text{uses}(u, t, p)$ is true if a user $u \in U$ uses a tag $t \in T$ in a post $p \in P$.

A sample Users - Tags network is given in Figure 5.8 to demonstrate the relations. This figure also shows the relations between users based on commonly used tags. The edges between users and tags show that they are associated to the interest areas represented by these tags. The figure also shows that there are common tags used by more than one user, and tags which are only used by a single user. The weights of the edges also demonstrate the strength of relations between users and the interest areas represented by tags they are associated to.

In the next step of network formation of this model, the Related Words network and the Users – Tags network is combined to model the indirect relations between the users, tags and words. Details of the Users – Tags – Words network are explained in the next section.

5.2.4. Interest Based User Networks

This network represents the set of users who are connected based on an area of interest. The model do not use any explicit information (such as mentions, replies) about interactions between users. Therefore, we aim to relate users to each other solely according to their contributions. Relations between users allow us to;

- find users who do not have any relationship, even not aware of each other but they have common interest. Finding such people provides a way to suggest them to follow each other or interact. In our method, suggestion part is not implemented but proposed as one of the aims of this method.
- discover interest specific communities since all the relations are based on contributions of users and the users are associated to each other based on their common interests.

A sample representation of this type of network is shown in the Figure 5.10.

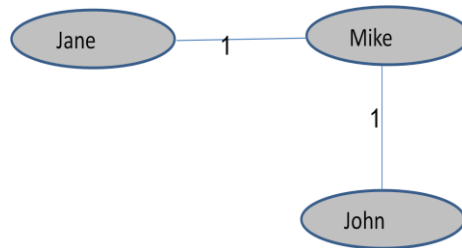


Figure 5.10. Sample Interest Based Users Network

An Interest Based Users network is defined as a graph M with a set of nodes U and undirected edges E_{uu} .

$$M = (V_{uu}, E_{uu}) \quad (5.15)$$

The set of all nodes in the Interest Based Users network of a given set of users is represented as $v \in V_{uu}$.

$$V_{uu} = \{ v_{uu} \mid v \in U \} \quad (5.16)$$

In Interest Based Users network, if there is an edge between user $u_i \in U$ and $t_x \in T$ and also, there is an edge between user $u_j \in U$ and tag t_x then, the edge between users u_i and u_j is constructed. The set of edges in an Interest Based Users network is defined as:

$$E_{uu} = \{(u_i, u_j) \mid (u_i, u_j) \in U \wedge (\exists t_k \in T: ((u_i, t_k) \in E_{ut}) \wedge ((u_j, t_k) \in E_{ut})) \wedge (\exists \text{weight} \in N^*: \text{weight} = \text{commonFreq}(u_i, u_j))\} \quad (5.17)$$

where

$$\text{commonFreq}(u_i, u_j) = |\text{commontags}(u_i, u_j)| \quad (5.18)$$

which defines the frequency of the tags that two users $u_i, u_j \in U$ use in common

$$\begin{aligned} \text{commontags}(u_i, u_j) = \{ \{t_1, \dots, t_k, \dots, t_n\} \mid u_i, u_j \in U \wedge e_{ik}, e_{jk} \in E_{ut} \} \\ 0 \leq n \leq |T| \end{aligned} \quad (5.19)$$

The assumption here is that users who have common interests tend to use the same tags. The more two users use the same tags in their microblogs, the more they are likely to share similar interest areas.

A sample network for a sample user “zef” is given in the Figure 5.11 which shows user relations based on the common tags they use. Previous network definition given in the Section 5.2.4 shows these common tags shared by users. Here, we simplify that network in order to focus on user relations and analyze social network properties of interest based user networks. Therefore the tags that users share are not displayed in this network.

During evaluation of the model, an application is developed to list all the common tags and other words related to these tags which are not displayed in interest based user networks.

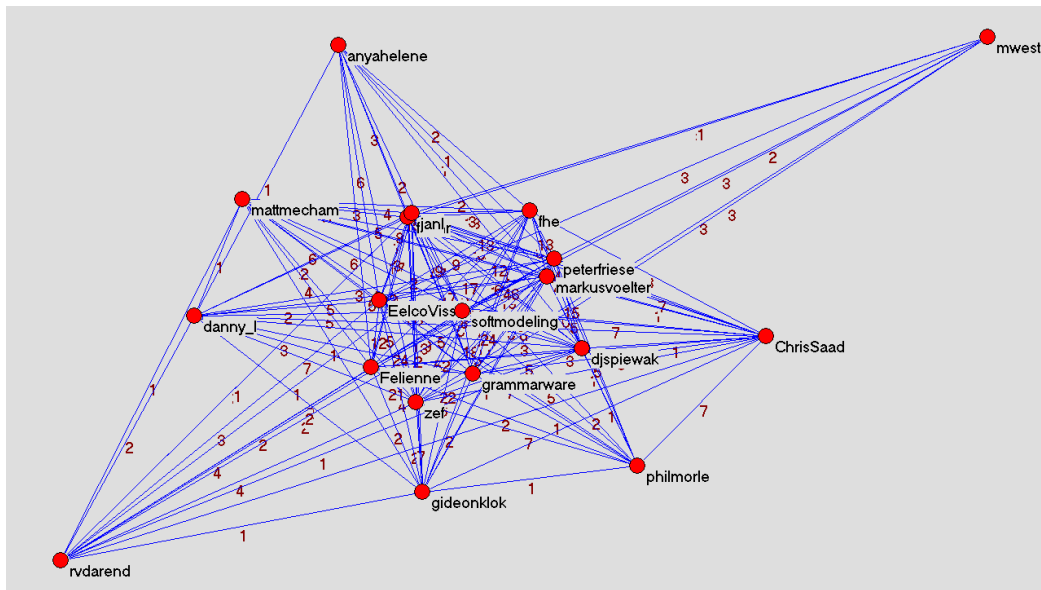


Figure 5.11. Users Network for a sample user

5.3. Semantic Grounding

The final step in our model is to map the words which are connected in our related words network to other knowledge bases such as DBPedia[84], Wikipedia[82] and Google[83]. We have three main objectives while mapping these words to the other resources as listed below:

- Cluster words and find an abstraction for the areas of interests which are defined as sets of related words in Section 5.2.1. Related words network is defined as a combination of sets of associated words where each set of related words represents an area of interest. These sets can be clustered under a category that is defined an abstraction for the interest area. By mapping a set of related words to Wikipedia, DBPedia and Google, we try to find a category that they can be grouped under. Wikipedia and DBPedia are more structured in order to find the hierarchies of concepts. However, formal definitions of the words are defined in general. Google on the other hand, provides search results and their categories for a vast set of words defined or not defined in any dictionary. The assumption here is that more than one word or tag represented in related words network can be grouped under a category defined in Wikipedia, DBPedia and Google. If none of words are found in any of the

resources, it is not possible to define an abstraction for those defined as interest areas.

- Find the meaning of the words which have special notations. Due to the shortness of the microblog posts, microbloggers frequently use short notations of words that make them difficult to understand. In general, a special notation, understandable by only a specific community of users, emerges. The resulting set of nodes in the related words network may possibly contain such words. They may possibly be used as tags as well. Such words can not be distinguish from tags and other words. By mapping a set of related words to Wikipedia, DBPedia and Google, to find a category that they can be grouped under, the words which are not defined in a dictionary but used by microbloggers can be understood by people who do not know the short notation and meaning of such words.
- Eliminate the words which can not be clustered under a specific category. Consider that a set of related words includes “java”, “eclipse”, “oop”, “write”. As a result of co-occurrence it is possible to find the word “write” with other words related to programming. In this case, eliminating the word “write” from the related words set that can be categorized under programming would improve the quality of set of related words.

In this phase we select the set of connected words and query DBPedia, Wikipedia and Google in respective order. If two words belong to the same category defined in DBPedia, we group them in the same cluster. If the word can not be found in DBPedia then we send a search query to Wikipedia similarly. Due to the nature of microblogging and the structure of the microblogs which are short text messages, there is a high possibility that we may not be able to map the words to either DBPedia or Wikipedia. For example, popular tags used in microblogging web site Twitter such as “#followfriday” or a word specific to a group of users “gtts” may not be found. In this case we refer to Google search API to map the words into a broader knowledge base. However, Google API returns titles of the pages containing the words instead of categories of the words. In this case, the common words in the titles are extracted to find an abstract definition common for all the words in the related set.

We have implemented mapping the set of related words to Wikipedia, DBPedia and Google as explained in the last step of our model. However, a further step which we left out of our scope in this thesis is required to combine all the results returning from all three resources. For example, most of the words are not found in Wikipedia or DBPedia due to the case sensitivity such as it is required to search for “iPhone” but not “iphone” though the concept iphone is defined in these resources. In many cases no result returned in terms of categories due to the specific notation of words in microblogs. When mapped to Google, the result set required to define a processing algorithm to find an abstraction for the category of common words in the result set. Therefore, we consider that an inference engine to combine and process these results would get over the mapping problems we face in the last step of our model.

In the following chapter, we explain in detail how we implemented our model in Java 2 Platform. Besides, we introduce the tools we used during the analysis of our results.

6. IMPLEMENTATION

In this chapter, the model proposed in Chapter 5 is explained in detail. The model is implemented to demonstrate it as a working example and evaluate the model based on social network analysis measures such as betweenness, closeness and degree.

In Section 6.1, Java 2 Platform [42] and Eclipse IDE [43] is introduced as our choice of implementation platform and the rationale behind our decision is provided. In Section 6.2 we describe Twitter API briefly and explain its functions that we call in our implementation. In Section 6.3, the functions coded to implement the model together with the limitations and constraints are explained. In Section 6.4, an overview of the network analyzing tool Pajek is given [44][45] and how it is used to evaluate our results is explained. In Section 6.5, we introduce MySQL [46] environment and explain the data structure of our implementation in MySQL database server. Finally in Section 6.6, the details of the data selection method and the selected data that we used to analyze and evaluate our implementation is described.

6.1. Implementation Platform

In this research, the proposed model is implemented in Sun Java 2 Platform version 5 [42] using Java programming language. Eclipse IDE [43] software development environment is used to edit, compile and debug the source code.

Our model does not require a language or platform dependent implementation. The only requirement for us was to implement our model aligned with Twitter Application Programming Interface (API) [48]. Though Twitter API supports many languages such as C#/.NET, Java, C++ etc [47], we chose Java 2 Platform among the two leading technologies Java 2 platform and Microsoft .NET Framework [49]. The main reason behind our decision to choose Java 2 Platform was our past experience with Java language and familiarity with the tools which support application development in Java. However, there are also other minor factors that effected our decision such as ease of available

features, tools and resources offered, library support and availability of free open libraries for the Java 2 Platform.

Eclipse IDE [43] is used to develop, debug and deploy the source code. It is a multi language software development environment and can be extended by adding new specific functionalities such as user interface development tools. Eclipse is a free and open source software which was the main rationale to use it as our development environment.

Here by, we would like to note again that our decisions on Java 2 Platform and Eclipse IDE are independent from the capabilities and architecture of our model.

In the next section, a brief overview of Twitter API is given and how is is integrated into the model is explained.

6.2. Twitter API

Application Programming Interface (API) is an interface implemented to allow a software application to interact with another. Similar to the interfaces that provide human and computer interactions, it allows interaction between applications [81]. An API is an abstraction of implemented functionality that it describes. APIs are implemented by writing function calls which provides other applications to invoke these functions. Calling conventions such as parameters, data structures, functions and protocols to invoke an implemented function are described in the API [81].

APIs allow software applications to interact with each other. In addition, they make working with other applications easier and allow them communicate across different computing platforms. The Internet and the related technologies, created the need for applications in different computing platforms to interoperate, exchange data, process tasks collaboratively and generate communities among users who share similar interests [50]. Service Oriented Architecture and its implementation, Web Services, emerge as a result of this need. However, current trends in web development are toward more direct Representational State Transfer (REST) methods[81].

In the context of web development, an API is defined as a set of request and response messages. The formats of these messages can be defined as HTTP [86], XML [87] or JSON [88]. APIs are widely used to develop web applications which combine functionalities of more than one external services, also known as mash-up applications [51]. Mash up applications use APIs of different sources and create a new service. They are associated with Web 2.0 technologies. There are different kinds of APIs available in the web such as mapping, video and photo, search and shopping and news [51]. Google Maps API [64], for instance, allow developers to combine any kind of data, such as real estate, with the location information. A REST API provides methods to request and get responses in the form of HTML, JSON or XML. In this implementation, the API Library Twitter4J[39], which a REST API implemented in Java is used to retrieve Twitter data.

Twitter exposes its data via its API. It consists of two parts: Search API and REST API. REST API functions allow developers to access Twitter data such as users tweets, status data and user information while search API allows developers interact with Twitter Search [65] and trends data. Due to the functionality limits of Search API, REST API is implemented in this work.

Twitter limits calls to the REST API by 150 requests per hour. However, 20,000 calls per hour are allowed for the authenticated API calls which are put into Twitter's white list upon developer's request. Throughout this research, we were in Twitter's white list so that we were able to inquire the data it exposes via its REST API.

In our implementation, the following methods defined in Twitter REST API is invoked;

- `account/rate_limit_status`: This method returns the remaining limit of a given account, which is authorized to query Twitter database. Since the number of requests is limited to 20,000 per hour, the remaining number is monitored during the data retrieval.
- `statuses/user_timeline`: This method returns the most recent twenty tweets posted by a given user. In our implementation, page parameter is used to retrieve past tweets of a

given user. By using paging, tweets are retrieved in the sets of twenty tweets per page. Therefore, this method is invoked by input parameters username and page number in order to retrieve all recent and past tweets of a given user.

In order to integrate the API method invocation with our implementation in Java 2 Platform, we use an open source Java library for the Twitter API: Twitter4j [39]. In Figure 6.1 source code of the implementation of Twitter API is given.

Twitter class in Twitter4J library is defined as the initial object to connect to Twitter API by passing the authentication parameters: user name and password. We invoke `getUserTimeline(username, page)` method of the Twitter object to retrieve all the tweets of a given user. This method returns the status updates, namely tweets, of a user. Instead of retrieving all the tweets of the user at once, it requires paging them by 20 tweets per page. Therefore, the API is invoked in a loop until all the tweets are retrieved.

The type of the returned value is a list of Status objects. Status class is a data class representing one single tweet of the user. It represents all the data of the tweet such as entry time, type (retweet, reply etc), user data and text. If a tweet is a reply to another user, we can easily get the unique tweet identification number (`tweet_id`) of the replied tweet and also the unique user identification number (`user_id`) of the replied user. We retrieve and use the data of user ids of the replied users while selecting our data set.

In Section 6.3, we describe the functions we have developed to implement our proposed model.

```

import twitter4j.Paging;
import twitter4j.Status;
import twitter4j.Twitter;
class RetrieveTweets {
    public void retrieveUserTweets(String inputUser)
    {
        Connection conn = null;
        Statement stmt = null;
        ResultSet rs = null;
        String tusername = "authenticatedUser";
        String tpassword = "password";
        Twitter twitter = new Twitter(tusername, tpassword);
        Class.forName("com.mysql.jdbc.Driver").newInstance();
        Paging page = new Paging();
        int i=1;int k=0;
        while (i!=0)
        {
            int limit =
twitter.rateLimitStatus().getRemainingHits();
            k++;
            page.setPage(k);
            List<Status> statuses = null;
            statuses = twitter.getUserTimeline(tusername, page);
            i=0;
            for (Status status : statuses) {
                i=1;
                stmt = conn.createStatement();
                DateFormat dateFormat = new
SimpleDateFormat("yyyy/MM/dd HH:mm:ss");
                java.util.Date date = new

String datetime = dateFormat.format(date);
                String qry=
                "INSERT INTO tweets
                (tweet_id,user_id,user_name,tweet,tweet_ti
                me,
                entry_date,isRetweet,inReplyToUser,
                inReplyToStatus,inquiryuser)
                VALUES
                ("
                +status.getId()+","
                +status.getUser().getId()+","
                +status.getUser().getScreenName()+','
                +status.getText()+','
                +status.getCreatedAt().getTime()+','
                +datetime+','
                +status.isRetweet()+','
                +status.getInReplyToUserId()+','
                +status.getInReplyToStatusId()+','
                +inputUser+')";
                stmt.executeUpdate(qry);
            }
            System.out.println("Remaining limit.."+limit+"\n");
        }
    }
}

```

Figure 6.1. Source code to retrieve Tweets using Twitter API

6.3. Implemented Functions

In Figure 5.1, the process flow of our model is shown in a single view. Here, we explain our implementation to realize these steps in detail.

In our model, the first step is to retrieve the tweets to process. RetrieveTweets class in our implementation has the method retrieveUserTweets(userName) which contains the getUserTimeline(username, page) API call to Twitter which retrieves all the tweets of a given user. getUserTimeline(username, page) method returns a list of objects of type Status class in Twitter API. As we described the Status class in Section 6.2 in detail, it includes all the data regarding the tweets of the given user. The list of Status objects which includes the tweets of the given user are inserted into the Tweets table in the database. retrieveUserTweets(userName) is invoked from the TwitterCommunityDetection() class.

In this research, we propose to pre-process and afterwards analyze all the tweets from a set of users. Selection of a set of users is required due to the availability and capacity of our processing resources. Ideally, it would not be possible to process all the tweets from all Twitter users using our limited resources. Hence, we decided to limit the number of users to process as an implementation choice due to our resource restrictions. However, our model has no constraints or limitations on the number of users or tweets to process.

To evaluate our model, random users, which are listed under a specific interest area in WeFollow web site are selected. Then, users who are replied by these selected users are ranked and the most replied 20 other users are selected. So the final user set includes users who are randomly selected and 20 other users who are known to be replied, in other words have conversations with those randomly selected. Although the basis of our data selection is reply relations, we could have selected the users who are in the friends list or followers list of the given user. Furthermore, we could have selected users randomly since our proposed model is not dependent on the relation types of the users. However, we used the reply relations of users as selection criteria for the sample data. We explain the rational behind our decision later in section 6.5 in this chapter.

The next step in our model is to process the tweets of the users who are replied by a specific user. After we retrieve the tweets from a set of selected users, we find the list of other users who are replied by each of the selected users. We rank the list and re-invoke the `retrieveUserTweets(userName)` method for each of the users on top 20 of the ranked list.

The next step is to process all the tweets we gathered to parse them into words and eliminate the stop words, links, conversational words etc. The types of the words we eliminate are defined in the Section 5.1.3. Here, we explain how we implemented this function. `WordProcess()` class has `processTweets()` method which is invoked from `TwitterCommunityDetection()` class. `ProcessTweet()` method takes user name or user id as input parameter and retrieves all the tweets of the given user together with the tweets of all other users who are replied by the given user ranked by number of times they are replied. Each tweet is parsed into its words and punctuations are removed from the words. In addition, words are marked in the database if they are stop words, links, mentions, replies, time related words, common words or emotional words. Finally, all the words are inserted into the Words table in the database. The output of the first part of our model is the set of words which are categorized by the word type. The words which are of type to be eliminated will be filtered out in the next steps. However, we do not delete the words that we eliminate in the next steps to keep the statistical information. Our analysis in the next steps is solely based on the list of concise words and tags which do not contain the words we define in the elimination step.

In Figure 6.2, the source code for finding replied users and processing tweets is shown.

The second part of our model is implemented by creating queries on the database. Once we have the words table containing all the words from all the users we selected, we create words networks and interest based user networks. Instead of processing all the words from all the users, we decided to cluster data due to our processing resource constraints. We clustered the users based on our decision for the data collection. A user and 20 other users who are replied by the original user are considered in the same cluster. For each cluster, we generated a contribution network of users and a words network.

We first select all the distinct words which are used as tag at least once in the Words table. Afterwards, we generate interest based user networks and related words networks. To generate interest based user networks, these tags are associated with the users who use them at least once. If two users use the same word as tag at least once, these two users are connected. The number of distinct tags they use is the weight of their connection. To generate the words networks, these tags are associated with words that they co-occur in the same tweet at least once. If a word co-occurs with a tag at least once, the tag and the word are connected. The number of tweets they co-occur is the weight of the connection in the words network.

There are a sequences of views and queries in the database to generate interest based user networks and words networks. The result sets of both type of networks are used to visualize and analyze the structure and characteristics of the networks. We use Pajek [45] network analysis tool to visualize and analyze the networks. Pajek accepts data in a specific text file format as input. Hence, we implemented functions to create Pajek input files from the contribution network of users and words network data in our database. In Section 6.4, we introduce Pajek and explain how we utilized it in our implementation.

The last part of our implementation demonstrates the mapping of the tags and words to other reference data sources which are Google, Wikipedia and DBPedia. By implementing this mapping, we aim to associate the meanings of the tags and words that we extracted by generating the words networks.

There are three classes for each of the resources. DBPedia class has queryDBPedia() method which invokes DBPedia API [32] by passing a query string written in SPARQL [33] query language. The query inquires DBpedia database by the given input search parameter which is tags or words in our model, and return a list of categories that the inquired tag or word belongs to. Our objective to retrieve the categories of the searched words is to find out the words which are under the same category. If we can map some words under a specific category and filter out others which are not under the same category, we would be able to refine our words network and extract more meaningful results in terms of specific interest areas.

```

public class TwitterCommunityDetection {

    public static void main(String[] args) {
        TwitterCommunityDetection tcd = new TwitterCommunityDetection();
        conn = DriverManager.getConnection("jdbc:mysql://localhost/twitter", "root" ,"");

        stmtTcd = conn.createStatement();
        stmtTcd.executeQuery("SELECT distinct(user_name) FROM tweets t");

        rsUsersTcd = stmtTcd.getResultSet( );

        while (rsUsersTcd.next()) // for each randomly selected user
        {
            inputuser = rsUsersTcd.next();

            String qryUsr = "SELECT inreplytouser, count( tweet_id ) AS say "+
                "FROM tweets " +
                "WHERE user_name = '"+inputuser+"' and inreplytouser <> -1 "+
                "GROUP BY inreplytouser "+
                "ORDER BY say DESC limit 20";

            stmt.executeQuery(qryUsr);
            ResultSet rsUsers = stmtTcd.getResultSet( );

            while (rsUsers.next()) // for each user replied by randomly
                selected user
            {
                String processUsername = rsUsers.getString("inreplytouser");

                WordProcess wordProcessUsr = new WordProcess();
                wordProcessUsr.processTweets(inputuser,processUsername);
            }
            rsUsers.close ( );
            stmt.close ( );
        }
        rsUsersTcd.close ( );

        stmtTcd.close ( );
        conn.close();
    }
}

```

Figure 6.2. Source code for finding replied users and processing their tweets

Sample SPARQL query for the word “iPhone” is given below:

```
PREFIX p: http://dbpedia.org/property/
PREFIX rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
PREFIX skos: http://www.w3.org/2004/02/skos/core#
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
SELECT * WHERE { ?subject rdfs:label 'iPhone'@en. ?subject skos:subject
?categories. }
```

Figure 6.3. Source code for a sample SPARQL query for the word “iPhone”

```
QueryExecution qexec = QueryExecutionFactory.sparqlService(
"http://dbpedia.org/sparql", query);
```

Figure 6.4. Source code for API call to DBpedia

DBpedia API returns a `ResultSet` object which contains the set of categories of all resources with the label “iPhone” in the example below. The categories returned from the query for the example is also given in Table 6.1.

In the case that a word can not be found in DBpedia, we search for the word in Wikipedia and Google in the same way we do for DBpedia. However, they all have different APIs and different call structures. Therefore, we implement Google and Wikipedia classes separately and invoke their APIs respectively.

```
"http://en.wikipedia.org/w/api.php?action=opensearch&search=iPhone"
```

Figure 6.5. Source code for API call to Wikipedia

```
"http://ajax.googleapis.com/ajax/services/search/web?v=1.0&q=iPhone"
```

Figure 6.6 Source code for API call to Google

Table 6.1. DBpedia result set

Categories ResultSet for the word “iPhone”
http://dbpedia.org/resource/Category:2007_introductions
http://dbpedia.org/resource/Category:Touchscreen_mobile_phones
http://dbpedia.org/resource/Category:Apple_Inc._mobile_phones
http://dbpedia.org/resource/Category:Apple_personal_digital_assistants
http://dbpedia.org/resource/Category:iPhone
http://dbpedia.org/resource/Category:iTunes
http://dbpedia.org/resource/Category:Smartphones
http://dbpedia.org/resource/Category:Wi-Fi_devices
http://dbpedia.org/resource/Category:Digital_audio_players
http://dbpedia.org/resource/Category:iPod
http://dbpedia.org/resource/Category:Multi-touch
http://dbpedia.org/resource/Category:Portable_media_players
http://dbpedia.org/resource/Category:Cloud_clients
http://dbpedia.org/resource/Category:iPhone_OS
http://dbpedia.org/resource/Category:Personal_digital_assistants

The results return from Wikipedia API and Google API are in JSON object format. We implement openSearch method of Wikipedia API in our query parameter string as shown in the example above. OpenSearch method executes an incremental search function which causes to retrieve other similar words to be found as well. To avoid the irrelevant matchings that return from OpenSearch, we apply similarity metrics to the titles of the returned categories. We get the title of each category returned from Wikipedia API and apply Levenshtein similarity metrics to them. The words which are below the threshold 0.5 are eliminated from our result set in our implementation.

Google API does not return any categories but instead it returns titles of the web pages which are inquired as search string parameter. We implemented Google class to map the words which we have no response from DBpedia or Wikipedia. While Google returns response to any kind of queries by words, in Wikipedia and DBpedia, the response is only

returned for the predefined words and concepts in these two resources. Since not every concept is defined in Wikipedia and DBpedia yet, we also inquire Google resources.

In this section, we explained the implementation for all parts of our model. In the next step, we introduce Pajek and describe how we analyzed and visualize the networks we generated from the users-tags and tags-words associations in our dataset.

6.4. Pajek

Pajek [45] is a free software for the analysis of large networks. In our model, we map the relationships between users, associations among words and relations between users and words are mapped to the networks. These networks consist of thousands of nodes and edges. Pajek offers some approaches to analysis and visualization of such networks [44].

Pajek uses six data structures to implement its algorithms [45].

- network – main object (vertices and lines);
- permutation – reordering of vertices;
- vector – values of vertices;
- cluster – subset of vertices (e. g. one class from partition);
- partition – tells for each vertex to which cluster the vertex belongs;
- hierarchy – hierarchically ordered clusters and vertices.

In our implementation, we created network and partition data structures of Pajek to provide input data. Network data structure is presented as a text file with the extension “.net”. It contains the definition of all the nodes, so called vertices and edges between these nodes. Partition data structure on the other hand, is presented as a text file with the extension “.clu” and contains the information regarding the cluster each vertice belongs to. CreatePajekNetwork class in our implementation has the following methods to create network files with “.net” extension and partition files with “.clu” extension.

insertWordVerticesConciled(username): This method retrieves the distinct set of tags and words in the words network. All the words and tags are inserted to the Vertices table in

the database with a unique number assigned to them. The cluster types of the tags are set to 1 and words are set to 0 in order to be able to generate partition files later.

`insertUserVertices(username)`: This method retrieves the list of distinct users who use at least one tag in their tweets. All the users are inserted into the Vertices table with a unique number assigned to them. The cluster types of the users are set as 2 in order to be able to generate partition files later.

`insertTagUser(username)`: This method executes the query to generate the users and tags network which shows the edges between users and tags. The users and tags are represented with their unique vertex numbers in the Vertices table. The edges that are found in this method are inserted into the Pajek file in the database to be used to create “.net” text files.

`insertTagWordConciled(username)`; This method executes the query to generate the tags and words network which shows the edges between tags and words. The tags and words are represented with their unique vertex numbers in the Vertices table. The edges that are found in this method are inserted into the Pajek file in the database to be used to create “.net” text files.

`createPajekNetFileConciled(username)`: This method retrieves the vertices and edges in the Vertices and Pajek tables respectively. Then it creates the “.net” network file which is the input data structure for the Pajek application.

`createPajekCluFile(username)`: This method retrieves the Vertices table and gets the types of the nodes. Then it creates “.clu” partition file which is the input data structure for the Pajek application.

Sample of “.clu” and “.net” files are given in Figure 6.3.

.Net File	.Clu File
*Vertices 9	*Vertices 9
1 "rdfa" 0	0
2 "html5" 0	0
3 "data" 0	1
4 "linkeddata" 0	1
5 "semanticweb" 0	0
6 "web" 0	0
7 "linked" 0	2
8 "tutorial" 0	2
9 "semantic" 0	2
*Edges	
1 3 5	
1 4 6	
4 5 10	
5 9 4	

Figure 6.7. Sample Pajek input data

After generating the data files for Pajek, we utilized the Kamada-Kawai [52] algorithm for automatic layout generation and the Separate Components option which optimizes each component separately and tiles components at the end. This layout allows us visualize the vertices and the connections between them. In addition we can easily distinguish the connected and disconnected vertices in our data set.

In summary, we utilize Pajek to analyze the users and words networks which we extract as a result of our implementation. While doing this we use its visualization algorithms and network metrics measurement tools.

In the next section we introduce MySQL and explain our data structure and database implementation on MySQL.

6.5. Database

MySQL is relational database management software which provides server and client side facilities to query, edit, administrate and manage large amount of databases. It is widely used by Web 2.0 applications. The reason to choose MySQL as our database

system is our experience with it. Besides, the familiarity of the resources with the free web based administration interface also effected our decision.

We created the following tables and views in the database.

- Tables:
 - (i) Tweets: This table stores all tweets retrieved from Twitter. The key fields in the table are user_name, user_id, inReplyToUser, tweet_id
 - (ii) Words: This table stores all tokens processed after the tweets are parsed and categorized. The tokens in this table can be queried by their categories such as tag, link, emotional words etc.
 - (iii) Vertices: This table stores the vertices defined for each network to be visualized in Pajek.
 - (iv) Pajek: This table stores edges defined for each network to be visualized in Pajek.
 - (v) UsersNetwork: The result set of the query UsersNetwork returns the relations between users and the common tags they use. This result set is directly used as input for Pajek to generate interest based user networks. The results for each set of users, which is for each network, is stored in a table named as UsersNetwork_Username. The data is clustered in separate tables in order to retrieve each network later without the need to handle too many data processing among many records.
 - (vi) WordsNetwork: The result set of the query WordsNetwork returns the relations between tags and other words. This result set is directly used as input for Pajek to generate related words network. The results set for each set of users, which is for each network, is stored in a table named as WordsNetwork_Username. The data is clustered in separate tables in order to retrieve each network later without the need to handle too many data processing among many records.
 - (vii) StopWords: The stop words defined in English are stored in this table. The data is used during categorization of raw set of tokens. If a token is found in this table, it is categorized as a stop word.

- Views

- (i) UsersNetwork: This view combines a sequence of other queries which inquires Words table resulting in a set of user1-tag-user2 data set. The result set is stored in the UsersNetwork table. The data in the result set is given as input for the Pajek to generate interest based user network of the users in the dataset.
- (ii) WordsNetwork: This view combines a sequence of other queries which inquires Words table resulting in a set of tag-word-frequency data set. The result set is stored in the WordsNetwork table. The data in the result set is given as input for the Pajek to generate interest based user network of the users in the dataset.

7. EVALUATION

In order to evaluate our model we retrieved a set of data from Twitter. Based on our examination of microbloggers (see Chapter 4), we observed that users who interact are likely to be:

- *human users* as opposed to bots or spammers, and
- *active users* that publish content instead of passive users who mostly read them.

These criteria were significant for selecting our data set, since we wanted to examine human users who publish about domain-specific content that can be analyzed. The more user content the better we are able to assess its nature and how strongly it relates to other's content.

First, we selected 50 seed users from the Wefollow [21] web site, which ranks popular Twitter users according to specific categories. Then, for each of these users we selected the top 20 users they replied to resulting in 50 sets of 21 (counting the seed user) users each. The data collected is categorized in Table 7.1. Note that due to minor issues during data retrieval the total number of users was less than planned.

Table 7.1. Properties of data set in terms of frequencies

Users	Tweets	Words	Stopwords	Filtered words
923	2,040,394	30,388,034	17,890,288	12,997,746

For each user set, two networks are generated: users-tags-words network and interest based user network. Interest based user networks consist of user nodes, which are related by similarity of contributions. Tag use behavior is used to generate interest based user networks. However, the tags themselves are not presented. These networks expose how similar users are in terms of their contributions, but don't reveal anything about the interest itself. In order to expose the nature of a similar interest between users as well as its strength the user-tags-words network is utilized. In this case, users are related to the tags

they share and tags are related to the words they co-exist with words. A simple application was implemented to display the tags and the words/tags that they are associated with for a pair of connected users (see Figure 7.6).

Each network is generated from the seed user. It contains all the users that the seed user replied to and all the words and tags within the posts of these users. Each such network is referred to by the seed user's screen name.

For each user set an interest based user network was generated. The basic network properties (centrality, betweenness and degree) of the resulting networks were computed Figure 7.2 shows a sample results. The complete set of results can be found Appendix B.

Table 7.2. Evaluation of data

Network	Avr. Centrality	Avr. Degree	Avr. Closeness	Avr. Betweenness
CaptSolo	4	0.21324	0.28028	0.02021
alexlindsay	11	0.05882	0.09635	0.00171
joewalker	1	0.38462	0.43719	0.14204

The interest based user networks are evaluated based on centrality measures which are closeness, betweenness and degree in the social network analysis. Furthermore, the central nodes of each network have been extracted. While measuring centrality, weights are considered. Weights in interest based user networks represent the number of common tags used by different users. Hence, higher centrality values for nodes imply that a user has many common interests with other users. The aim to measure these metrics is to;

- Compare interest based user networks based on centrality measures
- Determine whether the users can automatically be related based on the content they share Explore patterns between users and network properties – are the users who contribute more are in the center of the network?
- Examine if determining central nodes help enable us to extract communities of common interests – strongly connected users can be identified by setting thresholds

for the number of common tags users share (edge weights in interest based user networks are the number of common tags)

The generated networks have demonstrated that without considering follower, friend, mention, and reply data, our model successfully determines interest-based communities. This method is applicable in any language and works with filtered post data. It connects users who post similar content and isolates users who do not have any content in common with any of the users in the network. A threshold can be set to isolate the users. In the sample interest based users network given in Figure 7.1, the threshold for the edge weights is set as five; therefore the lines with the value lower than five are removed in the sample network. This sample network is generated for the user “cforbesoklahoma” and other users replied by him.

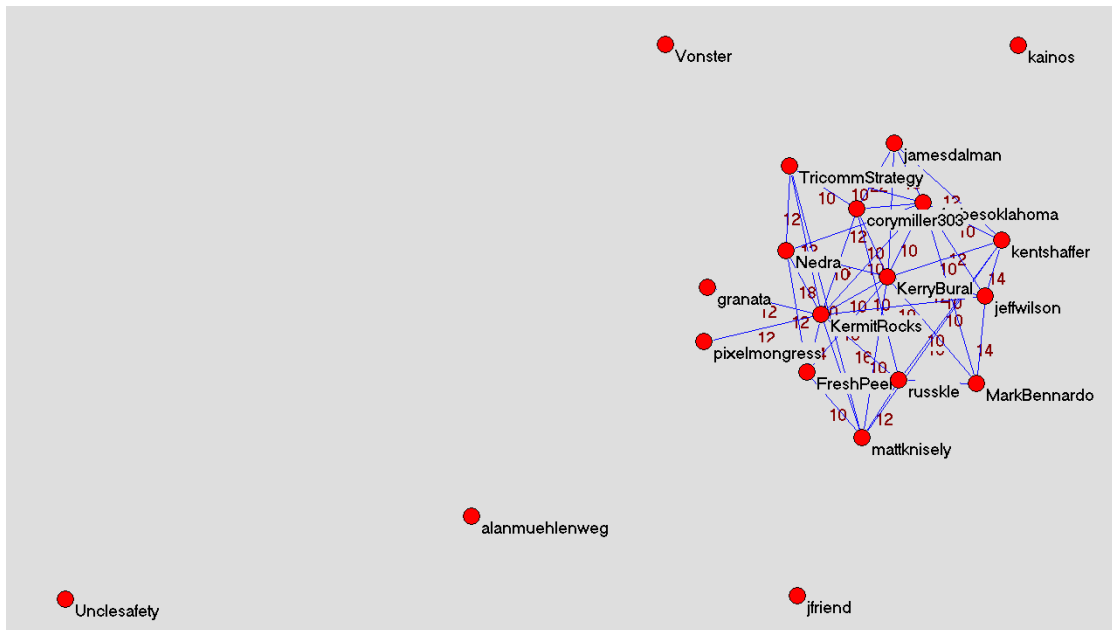


Figure 7.1. Connected and isolated users

We observed that the seed user is not necessarily one of the central nodes. This demonstrates that even though the user set was selected based on those replied to by the seed user, in the generated network users with strong similarity of contribution are the most significant. For example, in the interest based users network for seed user “appstoresocial” (see Figure 7.2), the user is not a central node (see Figure 7.3). This is

exactly what is desired since we are after discovering sets of users who actively contribute and converse about the same subject.

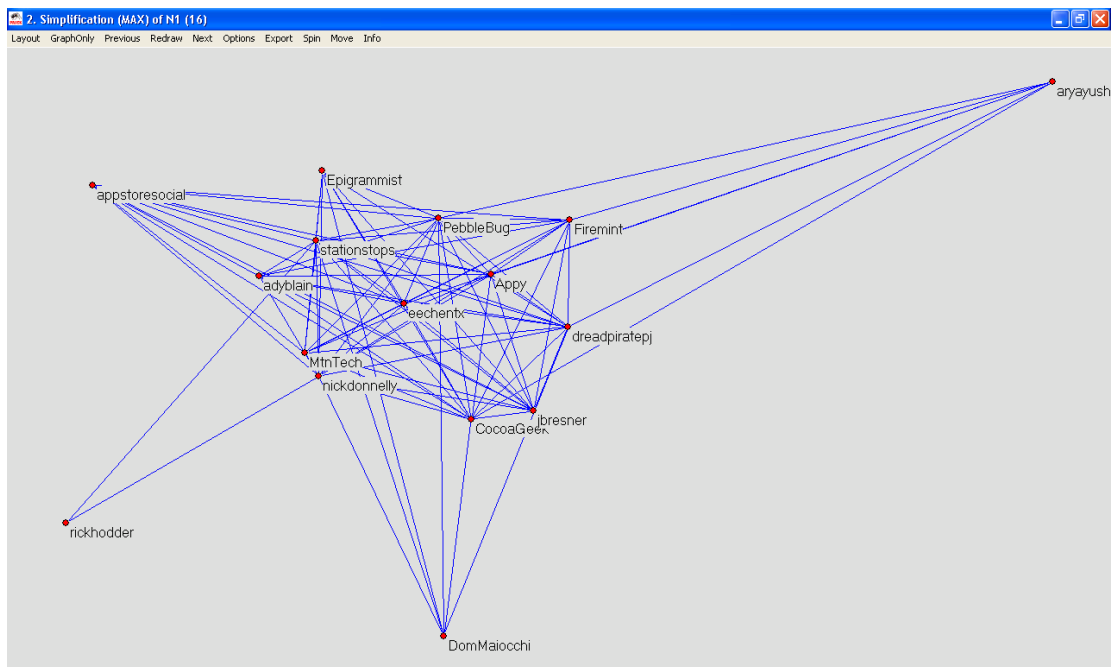


Figure 7.2 Interest Based User Network for a seed user who is not a central node.

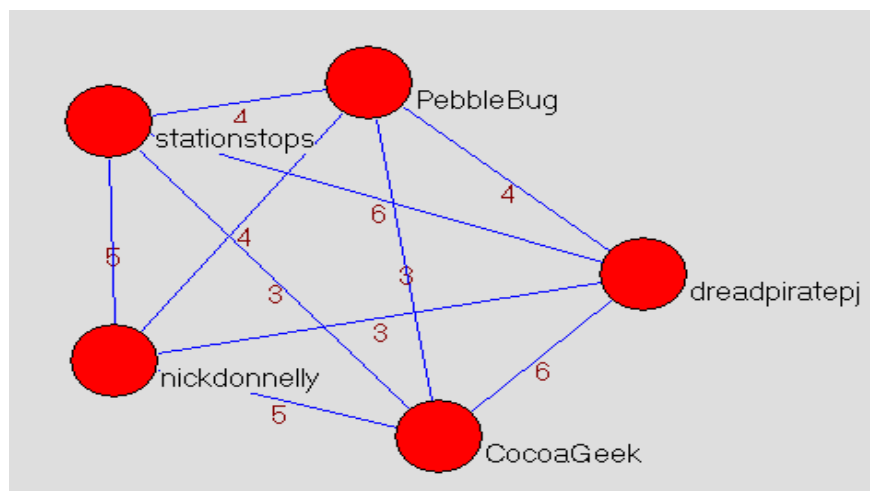


Figure 7.3. Central nodes of the seed user given in Figure 7.2.

The analysis of the interest based user networks, which are generated for each seed user, showed us that the centrality of the nodes changes from network to network. We observe that the networks with the lower values of betweenness, degree and closeness have

more central nodes. The ratio of central nodes to all nodes for all networks is given in Figure 7.4.

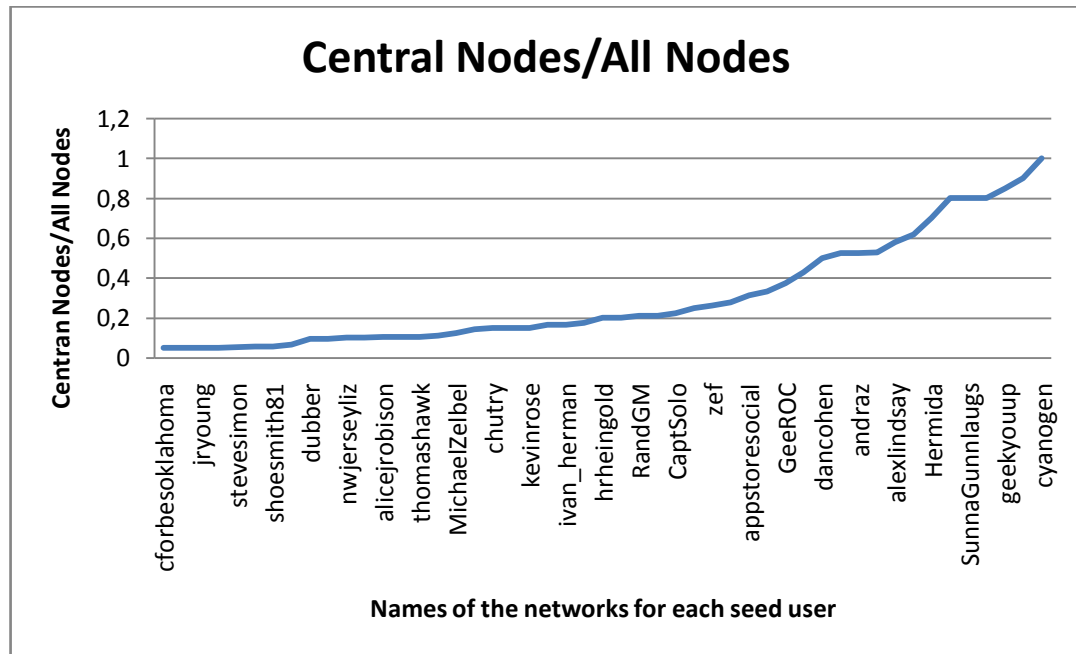


Figure 7.4 The ratios of central nodes/all nodes in all networks

Comparison of two types of networks with different number of central nodes is given in Table 7.2. The user “SunnaGunnlaugs” was selected as a seed user due to her interest in jazz music. Her profile shows that she is a popular microblogger with more than 1000 followers, post microblogs actively and has a wide range of different areas of interests such as jazz, foods, family and life in general. The other user in the selected example is “Ivan_Herman” who is an academic working on semantic web area and Linked Data [31] project. His microblogs are about semantic web as aligned with his area of interests. He posts specific microblogs about semantic web and developments in this area:

Table 7.3. Network measures comparison for two sample networks

Network	Centers	Avr. Degree	Avr. Closeness	Avr. Betweenness	User in the center?	Nodes
SunnaGunnlaugs	16 centers	0.01754	0.03098	0.00023	Yes	20
Ivan_Herman	3 centers	0.13235	0.20827	0.00832	No	18

The first network, “SunnaGunnlaugs” has 16 center nodes while the second network, “Ivan_Herman” has 3 centers. In Figure 7.4, the ratios of center nodes/all nodes are given. If the ratio of center nodes/all nodes converges to 1, it means that almost all nodes are central. We observe that in networks where this ratio converges to 1, all users have common interests but none of the relations are distinguished in terms of strength of the relation. The strength of the relation between two users is identified by the number of common tags they use.

Central nodes in the interest based user network for the seed user “Ivan_Herman” is given in Figure 7.5 and for the seed user “SunnaGunnlaugs” is given in Figure 7.7. The seed user “Ivan_Herman” is an active Twitter user who contributes about semantic web. The overall network that is generated for the seed user “Ivan_Herman” is given in Figure 7.6. It contains 18 nodes in total. However, only three of them are displayed as central nodes (See Figure 7.5). “Ivan_Herman” is not a central node although he is the seed user for this network and has conversations with other users. This shows that our model can identify user who have more common interest with many other users in the data set than “Ivan_Herman” has. We consider that these users are strongly connected in terms of common interests they share. User suggestion based on interest can be implemented by using these interest based relations.

On the other hand, the seed user “SunnaGunnlaugs” is another active user who contributes about jazz music. The interest based users network generated for the seed user “SunnaGunnlaugs” contains 20 nodes in total. Different from the network generated for the seed user “Ivan_Herman”, the number of central nodes is 16 which means that almost all the nodes are in the center of this network. The characteristic of this network is that almost all users have at least two or three common tags with many other users in the network. While the network generated for “Ivan_Herman” can identify significantly stronger relations between users, “SunnaGunnlaugs” type of networks can be used to extract communities of interests since this is a set of users most of which share similar tags (interests).

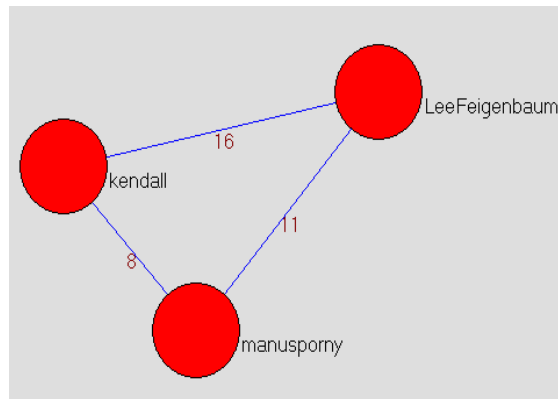


Figure 7.5. Central Nodes in an interest based users network with 3 central nodes

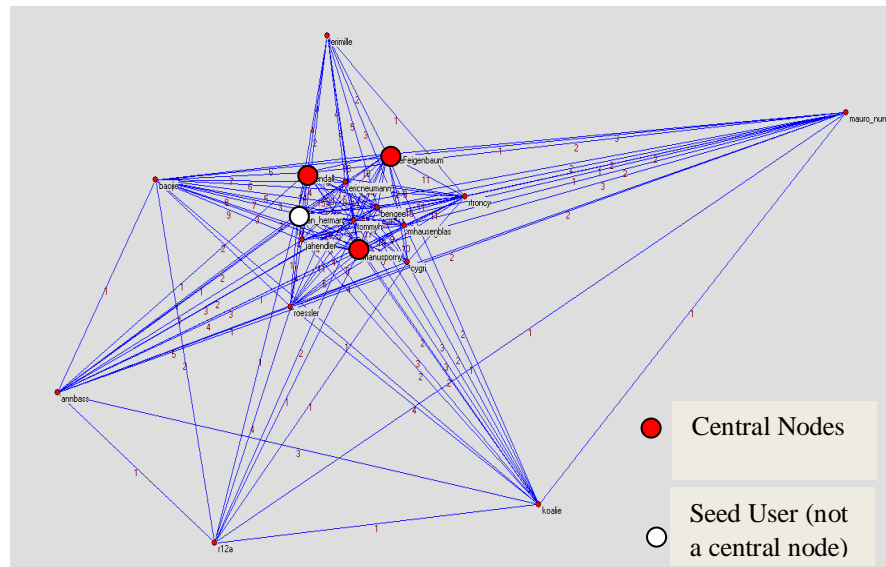


Figure 7.6. Overall interest based users network with 18 nodes(3 central)

In order to see the interest areas of the central users, the users – tags - words network is utilized. The list of tags and the words associated with the users “menusporny” and “kendall” (See Figure 7.5) is shown in the Figure 7.7. As proposed in our model, the common tags and other words related to these tags can be discovered. All the common tags “html5”, “linkeddata”, “semanticweb” and “rdf” used by “menusporny” and “kendall” are related to the area of semantic web. Therefore, we can observe that these two users have an area of interest in common regardless of knowing if they are already connected or not. If they are not, we can suggest one to the other based on the common interest they have. Furthermore, Figure 7.9 displays the list of related words associated to the tags used in common by the users “menusporny” and “kendall”. As tags are related to the area of

semantic web, we observe that associated words such as “rdfa”, “w3c”, “web” etc. are also highly related to the area of semantic web.

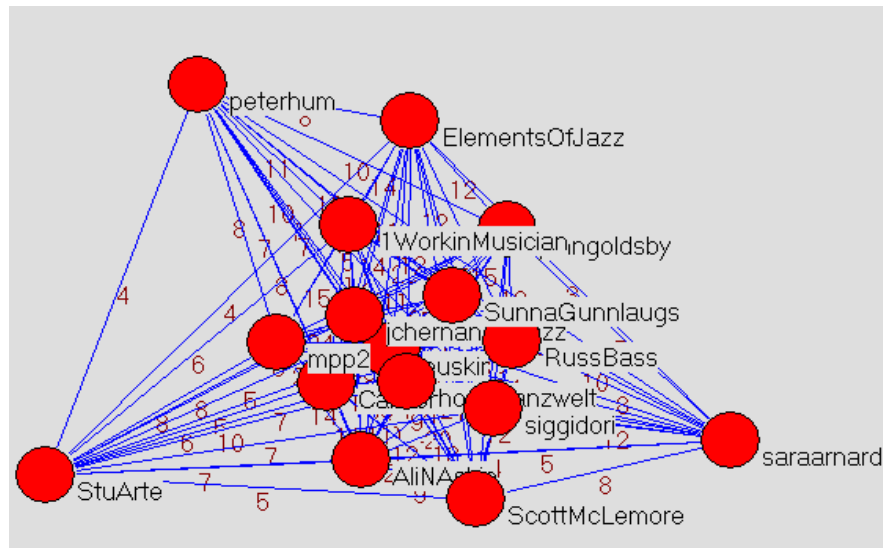


Figure 7.7. Central Nodes in an interest based users network with 16 central nodes

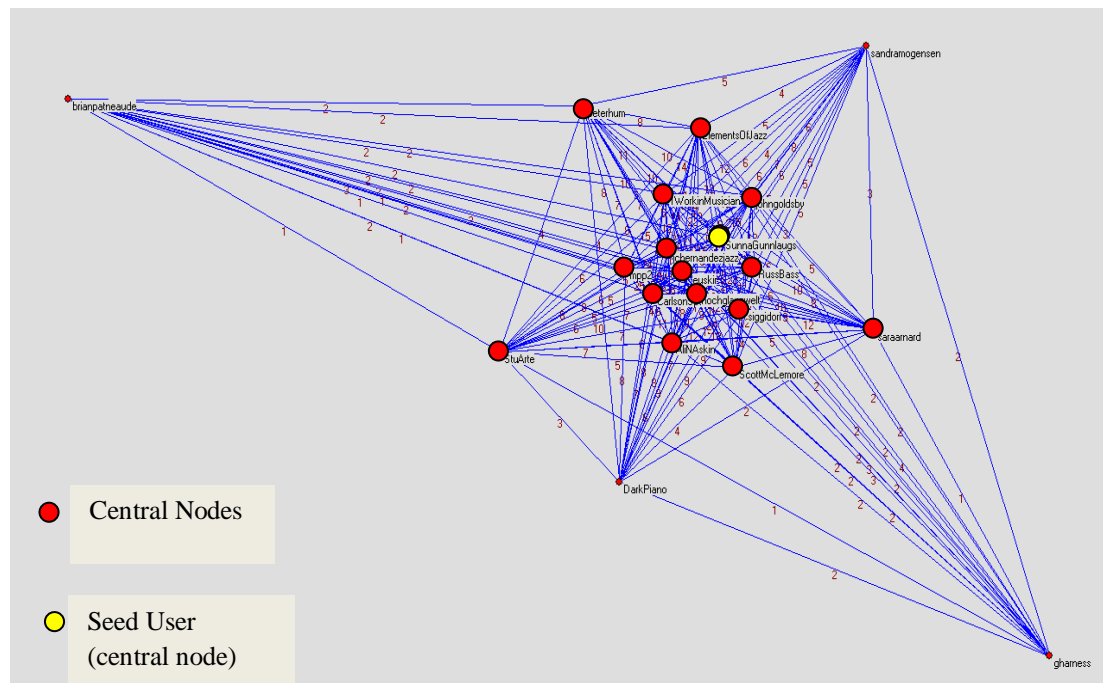
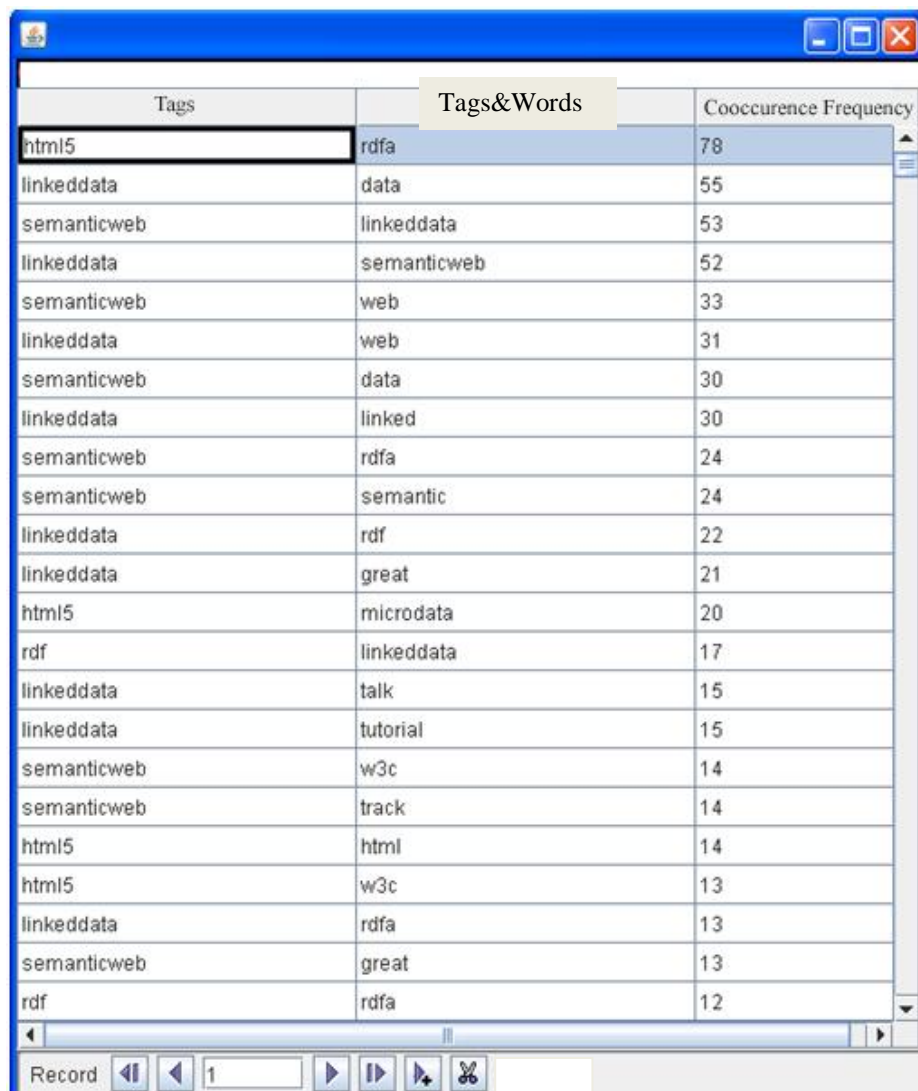


Figure 7.8. Overall interest based users network with 20 nodes (16 central).

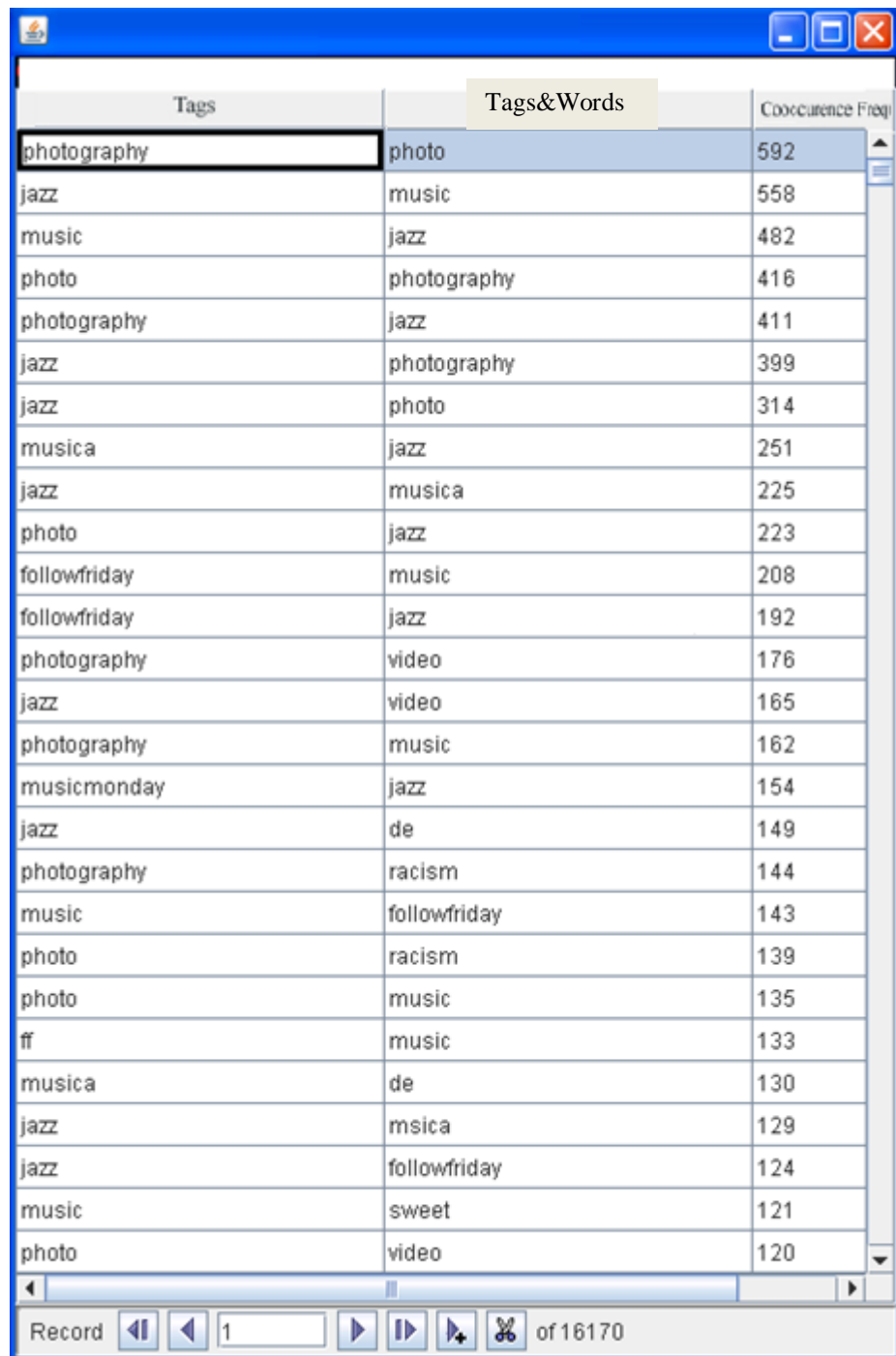
Since the tags are associated with other tags if they co-occur, the list shows the relations between tags as well such as “linkeddata” and “semanticweb”. They are listed both as a tag and associated word in the list.

In the second example network “SunnaGunnlaugs”, the keywords which are relevant to the interest area of jazz is extracted between connected pairs of users. The list of keywords associated with the users “jchernandezjazz” and “euskir” who are displayed as the central nodes in this network are given in the Figure 7.10. Again we can say that our model finds relations between people based on common interests and also extract these areas of interest people have in common.



Tags	Tags&Words	Cooccurrence Frequency
html5	rdfa	78
linkeddata	data	55
semanticweb	linkeddata	53
linkeddata	semanticweb	52
semanticweb	web	33
linkeddata	web	31
semanticweb	data	30
linkeddata	linked	30
semanticweb	rdfa	24
semanticweb	semantic	24
linkeddata	rdf	22
linkeddata	great	21
html5	microdata	20
rdf	linkeddata	17
linkeddata	talk	15
linkeddata	tutorial	15
semanticweb	w3c	14
semanticweb	track	14
html5	html	14
html5	w3c	13
linkeddata	rdfa	13
semanticweb	great	13
rdf	rdfa	12

Figure 7.9. Associated tags and words in a sample user network with 3 central nodes



Tags	Tags&Words	Cooccurrence Freq
photography	photo	592
jazz	music	558
music	jazz	482
photo	photography	416
photography	jazz	411
jazz	photography	399
jazz	photo	314
musica	jazz	251
jazz	musica	225
photo	jazz	223
followfriday	music	208
followfriday	jazz	192
photography	video	176
jazz	video	165
photography	music	162
musicmonday	jazz	154
jazz	de	149
photography	racism	144
music	followfriday	143
photo	racism	139
photo	music	135
ff	music	133
musica	de	130
jazz	msica	129
jazz	followfriday	124
music	sweet	121
photo	video	120

Figure 7.10. Associated tags and words in a sample user network with 16 central nodes

In the interest based user networks, generated by our model, where number of central nodes is high, users who are not connected in Twitter by the follower or friends relations

can be suggested to each other. This point can be implemented as a future work of our research.

In summary we have evaluated our model by measuring the centrality measures of each network. We show that the relations between people can be extracted by associating the content they publish. Besides, we show that a community of interest can also be extracted by using the central nodes in the networks. In addition, we demonstrate that our model is capable of expanding the keywords which are specific to an interest area without requiring users to use them. In other words, users are associated with the tags directly but our model can easily find other relevant keywords which may not be used by the users.

In this research, we have proposed and implemented a model to explore interest area specific communities in microblogging relevant words in areas of interests and identify users who share common interests. Although tags are used rarely in microblogging, communities of interests can be identified based on tags. Tags also help eliminating most of the irrelevant or insignificant words to an area of interest.

As we proposed in our model as the final step, we mapped the words to DBpedia, Wikipedia concepts and Google search items. Due to the large number of words and tags in our dataset, we selected only a few of them to map. The results showed us that a reasoning engine is required to evaluate the results since the category information that we propose to use in our model either is not available in these resources or returns disambiguated responses. We leave this part of evaluation as a future work. However, our implementation is able to map the given words to these resources without trying to infer semantics behind them.

8. CONCLUSION

In this research, we have proposed and implemented a model to explore interest area specific communities in microblogging environments. We have introduced the user generated content and its enabling technologies gathered under the concept of Web 2.0 technologies.

User generated content allow people share knowledge, communicate and collaborate via social web applications which gained impressive popularity in the last years. Among different kinds of social web applications such as networking (Facebook, MySpace), bookmarking (Del.icio.us, Citeulike), sharing videos and photos (Flickr), blogging (FriendFeed, Mashable, ReadWriteWeb) etc, we focus on microblogging application Twitter.

Twitter differs from the social networking sites since the relationships between its users are based on their interest areas. Users follow other users who they think share knowledge in a specific area. It also differs from other social web applications where users share resources such as Flickr due to its structure which do not allow many users collaboratively tag a resource which is a short text message - not an image or bookmark. However, it is a big problem today to find people who share similar interests in Twitter and provide valuable content among many irrelevant and conversational dialogs.

In our research we focus on discovering the interest area of users by processing and analyzing the content they publish. Furthermore, we explore relationships between people who share similar interests by extracting the common keywords and other related words associated with these words. By implementing our model, we extract the communities associated with a specific interest area and expand the tag cloud of the users so that they can be searched not only by the words they use but also other keywords relevant to this interest area.

In this research, we have proposed and implemented a model to explore interest area specific communities in microblogging relevant words in areas of interests and identify users who share common interests. Although tags are used rarely in microblogging, communities of interests can be identified based on tags. Tags also help eliminating most of the irrelevant or insignificant words to an area of interest.

8.1. Overview

In the first chapter, we have given brief information regarding our motivation for this thesis and provided an overview of our research.

Next in the second chapter, we have given the background information related to our work in detail. Basic concepts of social web and user generated content (UGC) have been described. In addition, main issues regarding the user generated content has been explained with references to the related work in this area. In addition, tagging and collaborative tagging behavior of users in social networks has been described briefly as a solution to solve the issues regarding the UGC. Finally, the microblogging and Twitter environment which we have chosen as test bed for our implementation in this research. In this chapter, we have also given brief information about Social Network Analysis.

In Chapter 3, we define our problem statement together with a sample scenario regarding the problem.

In Chapter 4, the results of our analysis which has been performed to understand the structure of social networks, how people behave in microblogging environments and characteristics of Twitter environment has been described.

In Chapter 5, we have proposed our model to explore interest area specific users and communities in microblogging environments.

In Chapter 6, we have presented briefly the implementation of our model that has been completed as part of this research.

In Chapter 7, we have presented the cases we have tested to evaluate our model together with the results for each case.

In Chapter 8, we have provided detailed information about the future work and a summary of our research and our contributions as conclusion of our work.

8.2. Contributions

The major contribution of this thesis is to the problem of discovery of users' interest areas and finding similar users who provide valuable content and information in a specific area in the online social networks, specifically in the microblogging environments. Our model is distinguished from other approaches to find users in these networks since the relations we discover is based on the users' contributions instead of the explicit information given by them such as location, biography, friends, followers, number of updates etc. Since our basis for extracting the relations is the content itself, we provide avoiding the possibility of finding people who declare that they are interested in a specific area but do not provide any valuable content in this area.

In addition to these, we expand the tag cloud of the users by discovering associations between keywords so that even if users do not use a specific word, they can be inquired by all relevant words in this area. Hence, similar people are matched by a set of relevant words instead of using exact keyword match.

8.3. Future Work

We have initiated our research to find people who are interested in a specific area by giving a specific keyword as input. However, due to the time and resource constraints we narrowed down the scope of our thesis to find the interest area of a given user and other users who share similar interest. In addition we focused on discovering;

- the common interests between users
- the users and communities who are similar to a given user in terms of a specific interest area.

Throughout this thesis, we see that there are different directions which we left as future work for our proposed model due to the time and resource constraints. Here by, we explain these alternative directions in detail.

Users who are interested in a specific area can be found by searching a specific keyword. By associating the relevant words, tags and users as we propose in our model, searching for users who are associated with the set of relevant words in a specific area can be found. This work on the other hand requires indexing and ranking as the search engines do.

Another direction would be to implement a semantic reasoning engine for the set of words we associate so that the meaning of the interest area would be extracted. It is also possible to cluster the words according to a classification or categorization algorithm so that specific thresholds can be discovered and the less relevant words in the words set could be eliminated.

Mapping the set of words we associated can also be mapped to pre-defined ontologies such as ConceptNet and WordNet to extract the relationships between them such as is-a, has-a relations. This would also add a semantic view to the group of words which we associated with each other.

Due to the resource limitations, we have not implemented the path finding algorithms in the users-tags-words network in our proposed model. Such an implementation would be an extension to our model and basis for a user suggestion application.

In our thesis we applied our model on a set of users which we know that they have reply relations. With this information in one hand, we tried to find the relations between users based on a specific interest area. However, applying further social analysis algorithm such as clustering would expose the communities and the relations between these communities based on our network models for users-tags-words. Our model can be implemented by processing all the users in a single network to discover communities as a future work.

Our model requires the processing of all the tweets from all the users. Hence, realistically it is not feasible to process all the data and index all the users together with the information regarding their interest areas. During the evaluation phase of this thesis, we have faced performance problems due to the processing of large sets of data. Therefore, alternative ways to get over the performance problems in the large scale should be considered.

APPENDIX A. Stop Words in English

Table A.1. Stop words in English

A	Come	he's	my	seeing	truly
a's	Comes	hello	myself	seem	try
able	Concerning	help	n	seemed	trying
about	consequently	hence	name	seeming	twice
above	Consider	Her	namely	seems	two
according	Considering	here	nd	seen	u
accordingly	Contain	here's	near	self	un
across	Containing	hereafter	nearly	selves	under
actually	Contains	hereby	necessary	sensible	unfortunately
after	corresponding	herein	need	sent	unless
afterwards	Could	hereupon	needs	serious	unlikely
again	couldn't	hers	neither	seriously	until
against	Course	herself	never	seven	unto
ain't	Currently	Hi	nevertheless	several	up
All	D	Him	new	shall	upon
allow	Definitely	himself	next	she	us
allows	Described	His	nine	should	use
almost	Despite	hither	no	shouldn't	used
alone	Did	hopefully	nobody	since	useful
along	didn't	how	non	six	uses
already	Different	howbeit	none	so	using
also	Do	however	noone	some	usually
although	Does	I	nor	somebody	uucp
always	doesn't	i'd	normally	somehow	v
Am	Doing	i'll	not	someone	value
among	don't	i'm	nothing	something	various
amongst	Done	i've	novel	sometime	very
An	Down	Îe	now	sometimes	via
and	downwards	If	nowhere	somewhat	viz
another	During	ignored	o	somewhere	vs
any	E	immediate	obviously	soon	w
anybody	Each	In	of	sorry	want
anyhow	Edu	inasmuch	off	specified	wants
anyone	Eg	Inc	often	specify	was
anything	Eight	indeed	oh	specifying	wasn't
anyway	Either	indicate	ok	still	way
anyways	Else	indicated	okay	sub	we
anywhere	Elsewhere	indicates	old	such	we'd

Table A.1. Stop words in English (continued)

apart	Enough	inner	on	sup	we'll
appear	Entirely	insofar	once	sure	we're
appreciate	Especially	instead	one	t	we've
appropriate	Et	into	ones	t's	welcome
Are	Etc	inward	only	take	well
aren't	Even	Is	onto	taken	went
around	Ever	isn't	or	tell	were
as	Every	It	other	tends	weren't
aside	Everybody	it'd	others	th	what
ask	Everyone	it'll	otherwise	than	what's
asking	Everything	it's	ought	thank	whatever
associated	Everywhere	Its	our	thanks	when
at	Ex	itself	ours	thanx	whence
available	Exactly	J	ourselves	that	whenever
away	Example	just	out	that's	where
awfully	Except	K	outside	thats	where's
b	F	keep	over	the	whereafter
be	far	keeps	overall	their	whereas
became	few	kept	own	theirs	whereby
because	fifth	know	p	them	wherein
become	first	knows	particular	themselves	whereupon
becomes	five	known	particularly	then	wherever
becoming	followed	L	per	thence	whether
been	following	last	perhaps	there	which
before	follows	lately	placed	there's	while
beforehand	for	later	please	thereafter	whither
behind	former	latter	plus	thereby	who
being	formerly	latterly	possible	therefore	who's
believe	forth	least	presumably	therein	whoever
below	four	less	probably	theres	whole
beside	from	lest	provides	thereupon	whom
besides	further	Let	q	these	whose
best	furthermore	let's	que	they	why
better	G	like	quite	they'd	will
between	get	liked	qv	they'll	willing
beyond	gets	likely	r	they're	wish
both	getting	little	rather	they've	with
brief	given	look	rd	think	within
but	gives	looking	re	third	without
by	go	looks	really	this	won't
c	goes	Ltd	reasonably	thorough	wonder
c'mon	going	M	regarding	thoroughly	would

Table A.1. Stop words in English (continued)

c's	Gone	mainly	regardless	those	would
came	Got	many	regards	though	wouldn't
can	gotten	may	relatively	three	x
can't	greetings	maybe	respectively	through	y
cannot	H	Me	right	throughout	yes
cant	Had	mean	s	thru	yet
cause	hadn't	meanwhile	said	thus	you
causes	Happens	merely	same	to	you'd
certain	Hardly	might	saw	together	you'll
certainly	Has	more	say	too	you're
changes	hasn't	moreover	saying	took	you've
clearly	Have	most	says	toward	your
co	haven't	mostly	second	towards	yours
com	Having	much	secondly	tried	yourself
	He	must	see	tries	yourselves
					z
					Zero

APPENDIX B. Interest Based Users Networks Results

Table B.1. Results for each of the Interest Based Users networks for 49 seed users

Network	Central Nodes	Avr. Degree	Avr. Closeness	Avr. Betweenness	Original user in the center? Yes/No	# of Nodes	Central Nodes/All Nodes	# of tweets	# of words (concise list)	# of words before elimination	# of stopwords
CaptSolo	4	0,21324	0,280280	0,02021	Yes	18	0,22	40082	280059	630580	350521
alexlindsay	11	0,05882	0,096350	0,00171	No	19	0,58	47616	304456	749351	444895
joewalker	1	0,38462	0,437190	0,14204	No	15	0,07	31152	204830	476411	271581
andraz	11	0,05263	0,091420	0,00085	No	21	0,52	51477	319360	758801	439441
zef	5	0,15033	0,229000	0,01003	Yes	19	0,26	28474	187495	424313	236818
Alok_Jain	2	0,17647	0,261860	0,01657	No	19	0,11	44592	256873	661616	404743
johngirvin	12	0,01282	0,022890	0,00016	Yes	15	0,80	33639	179285	464540	285255
thomashawk	2	0,26471	0,264710	0,02918	No	19	0,11	45341	279815	670156	390341
RandGM	4	0,17647	0,269590	0,01225	Yes	19	0,21	46313	319206	761418	442212
wyntonnmarsalis	6	0,07143	0,113100	0,00446	No	18	0,33	20797	137929	311257	173328
MichaelZelbel	2	0,34615	0,472760	0,06963	No	16	0,13	28917	164454	387697	223243
stevesimon	1	0,33333	0,450000	0,0738	No	19	0,05	35391	236026	576947	340921
SunnaGunnlaugs	16	0,01754	0,030980	0,00023	Yes	20	0,80	45106	288510	650046	361536
ivan_herman	3	0,13235	0,208270	0,00832	No	18	0,17	18343	122193	272889	150696
GMarketingGuy	18	0,00585	0,010820	0,00003	Yes	20	0,90	56169	323986	859704	535718
eburnette	4	0,25146	0,303270	0,04356	No	20	0,20	40251	245027	600667	355640
paul_burwell	3	0,15263	0,229370	0,22937	Yes	21	0,14	41428	246506	644657	398151
plymouthgooner	12	0,08242	0,096020	0,1405	Yes	15	0,80	33220	168191	428495	260304
crazybob	1	0,34559	0,400980	0,04964	Yes	18	0,06	44819	270679	669588	398909
appstoresocial	5	0,22857	0,284660	0,06337	No	16	0,31	28252	184482	450967	266485
geekyouup	17	0,0117	0,021120	0,00011	Yes	20	0,85	42763	265783	664672	398889
shoesmith81	1	0,42647	0,557850	0,18558	No	18	0,06	19757	109170	277627	168457
amuse	3	0,11765	0,176490	0,02066	No	18	0,17	48382	285835	710601	424766
tommyh	9	0,08333	0,130990	0,00389	Yes	17	0,53	39113	239410	534778	295368
kevinrose	3	0,22807	0,298880	0,03063	Yes	20	0,15	48521	272316	685424	413108
GeeROC	6	0,1619	0,203550	0,13129	Yes	16	0,38	28774	201374	438799	237425
cforbesoklahoma	1	0,23392	0,299440	0,0991	Yes	20	0,05	35381	209073	529257	320184
cyanogen	17	0	0,000000	0	Yes	17	1,00	26451	151545	389518	237973
dfazekas	4	0,21905	0,298360	0,12835	No	16	0,25	30989	219846	488102	268256
mcleod	3	0,19298	0,287860	0,01498	No	20	0,15	58487	352608	897985	545377
gylphi	2	0,13158	0,194010	0,02188	No	21	0,10	47096	331557	726103	394546
jeanburgess	3	0,15	0,220770	0,0327	Yes	17	0,18	35867	203577	516431	312854
alicejrobinson	2	0,14379	0,228030	0,00702	Yes	19	0,11	47640	323740	718965	395225
chutry	3	0,10526	0,155140	0,02806	No	20	0,15	52565	339430	831587	492157
markdeuze	1	0,32749	0,389630	0,11879	No	20	0,05	41602	371433	570242	198809
TreborS	2	0,24265	0,354860	0,02444	No	18	0,11	37048	247431	541210	293779
barbarahui	2	0,25146	0,340460	0,10402	Yes	20	0,10	48099	328209	761288	433079
GeorgeOnline	5	0,09559	0,151890	0,00505	No	18	0,28	47917	314771	737227	422456
nwjerseyliz	2	0,16959	0,246890	0,05016	Yes	20	0,10	50550	284145	724396	440251
dubber	2	0,12105	0,183270	0,02051	No	21	0,10	55383	329699	836086	506387
Hermida	14	0,01754	0,324600	0,0001	Yes	20	0,70	49903	358246	815419	457173
dancohen	10	0,05848	0,099590	0,00112	Yes	20	0,50	42592	300395	668901	368506
academicdave	9	0,06316	0,109350	0,00099	Yes	21	0,43	56041	343037	842017	498980
hrheingold	4	0,24561	0,292270	0,11297	No	20	0,20	49124	334116	751461	417345
jayrosen_nyu	11	0,04737	0,082530	0,00074	Yes	21	0,52	64317	448643	1047664	599021
paullev	1	0,23977	0,307110	0,10419	No	20	0,05	40633	279267	699732	420465
mwesch	4	0,26471	0,293180	0,08804	No	19	0,21	37001	258263	600760	342497
presidentgee											0
jryoung	1	0,32164	0,416060	0,1003	No	20	0,05	37986	254070	594417	340347
PRsarahevans	13	0,03684	0,063240	0,00071		21	0,62	59033	321395	837265	515870
TOTAL						923		2040394	12997746	30888034	17890288
MAX		0,426470	0,557850	0,229370			1,000000				
MIN		0,000000	0,000000	0,000000			0,050000				
AVERAGE		0,166655	0,230223	0,048006			0,303757				
STANDARD DEVIATION		0,110852	0,1319854	0,056279844			0,269109				

APPENDIX C. Results Summary Tables for Interest Based User Networks

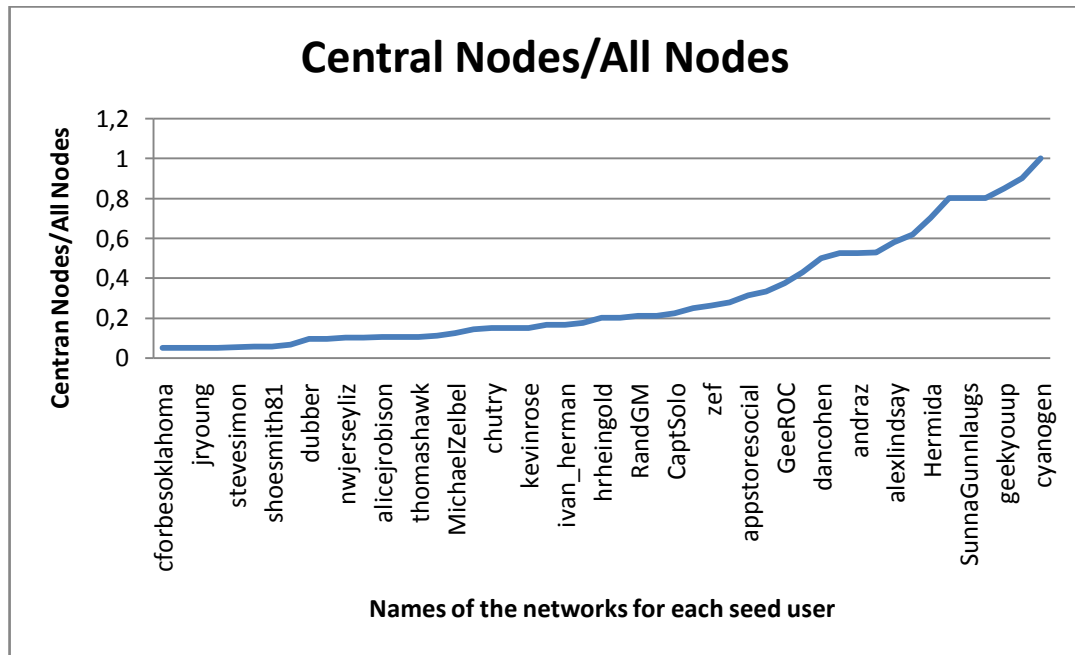


Figure C.1. The ratio of central nodes to all nodes in each interest based user network

Interest based networks that are generated for 49 seed users in the data set are analyzed to see the network properties they have. Central nodes of all networks are found in each network. Due that the size of the networks in terms of the number of nodes is different in each network, the ratio of central nodes to all nodes in each network is calculated to compare the characteristics of the networks. Figure C.1 shows the ratios of central nodes to all nodes in each network. The summary of the ratios given in Figure C.1 is given in Figure C.2.

In the selected dataset, 80 percent of the networks have ratios less than 0.6 (See Figure C.2). In our case, it means that there are significant users who have many common interests with many users in the network and those can be distinguished, using our model, based on the common interest they share with other users in the network.

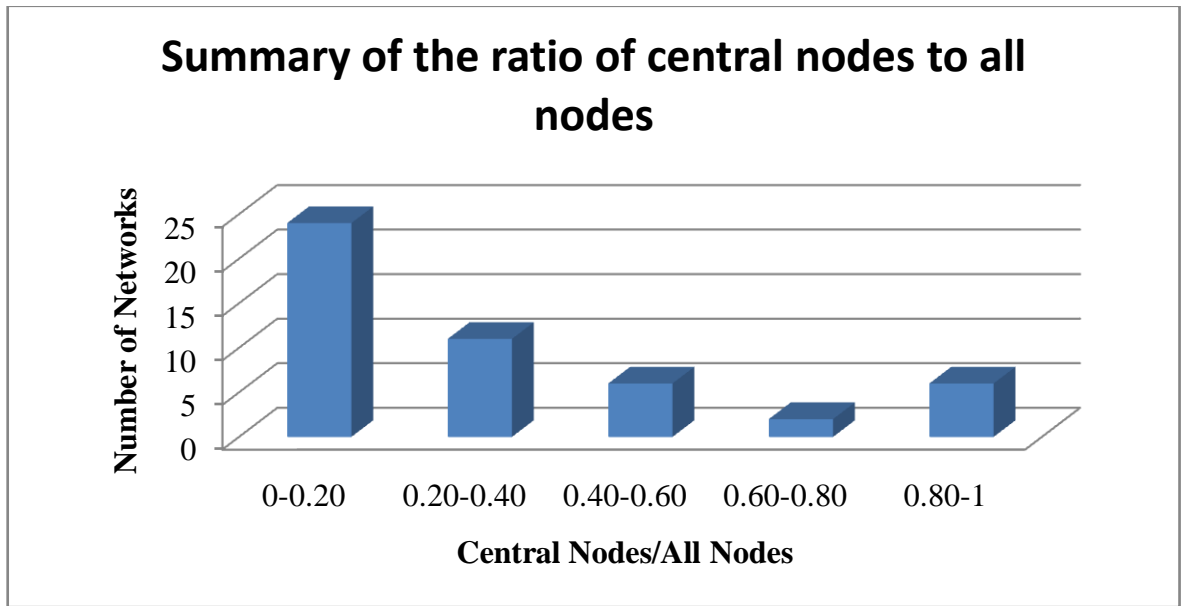


Figure C.2. The ratio of central nodes to all nodes in each interest based user network

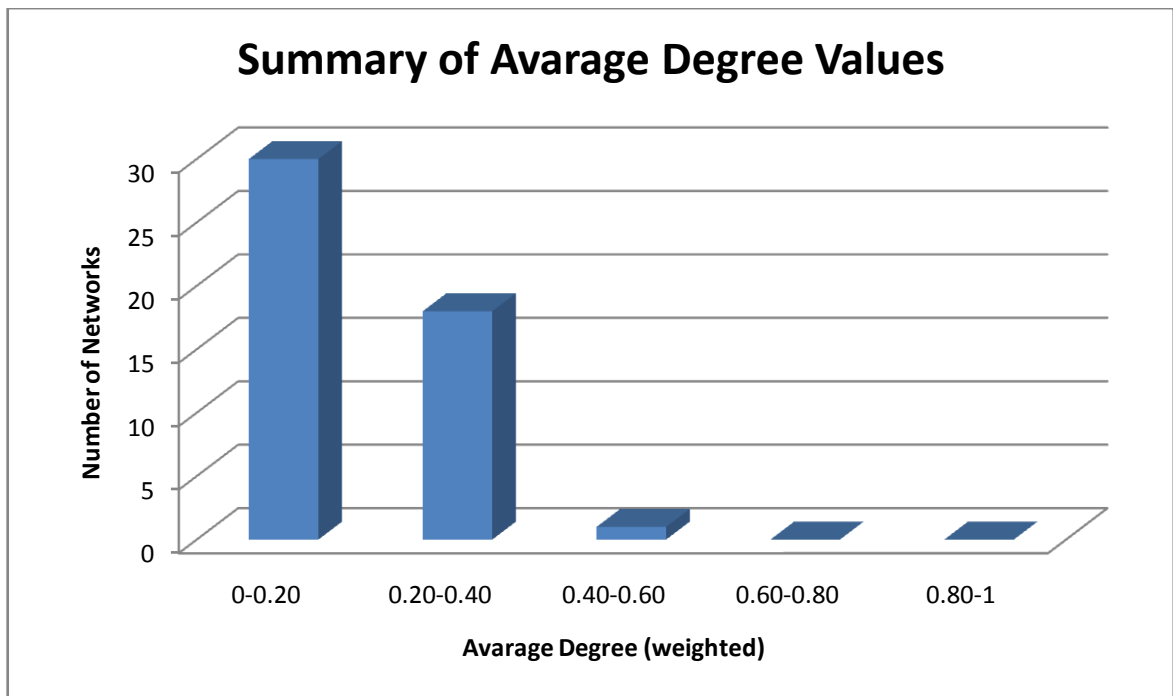


Figure C.3. Summary of average degrees

Figure C.3 shows the average degree distribution in networks. Weights are included to calculate average degrees. The average degrees of the networks are less than 0.5. This shows that even though there are central nodes in networks, all networks have nodes (users) with low degree values (share less or no common interests with other users).

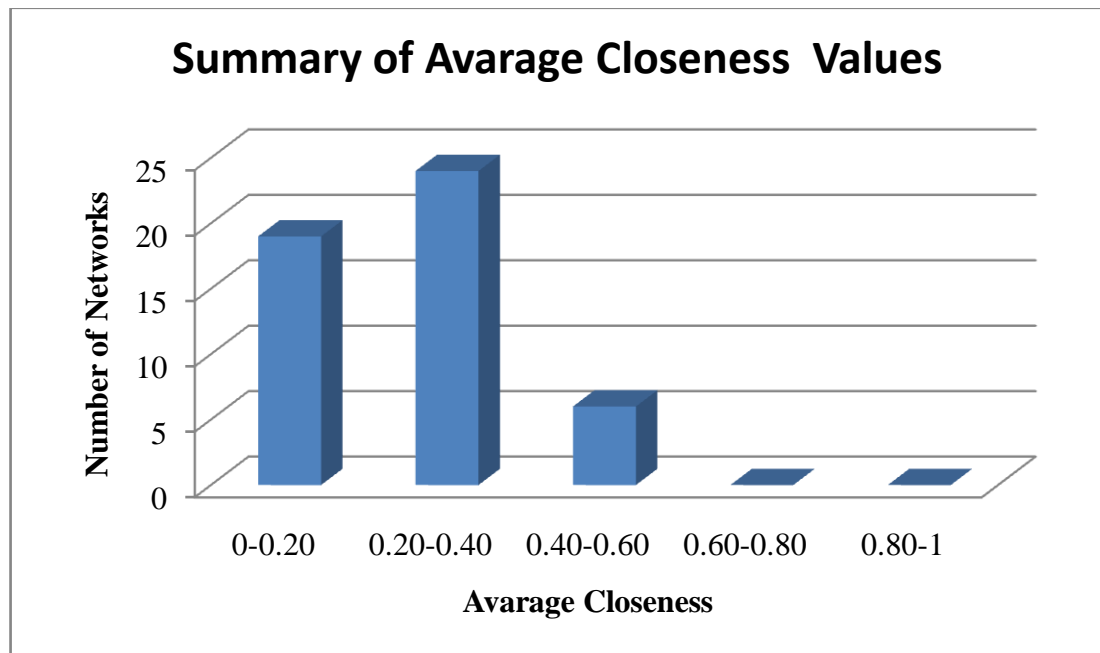


Figure C.4. Summary of avarage closeness values

Avarage closeness and betweenness values of the networks are given in Figure C.4 and Figure C.5. We observe that ten percent of the networks have closeness values between 0.4 and 0.5 and the rest is less than 0.4. Higher values of closeness and betweenness show that the networks have many users who can be reached in the graph. In our case, user suggestion can be studied in networks with higher closeness values. Betweenness values on the other hand are between 0-0.2 in all networks. This shows that none of the networks have all users connected to all others so that the average betweenness would be higher.

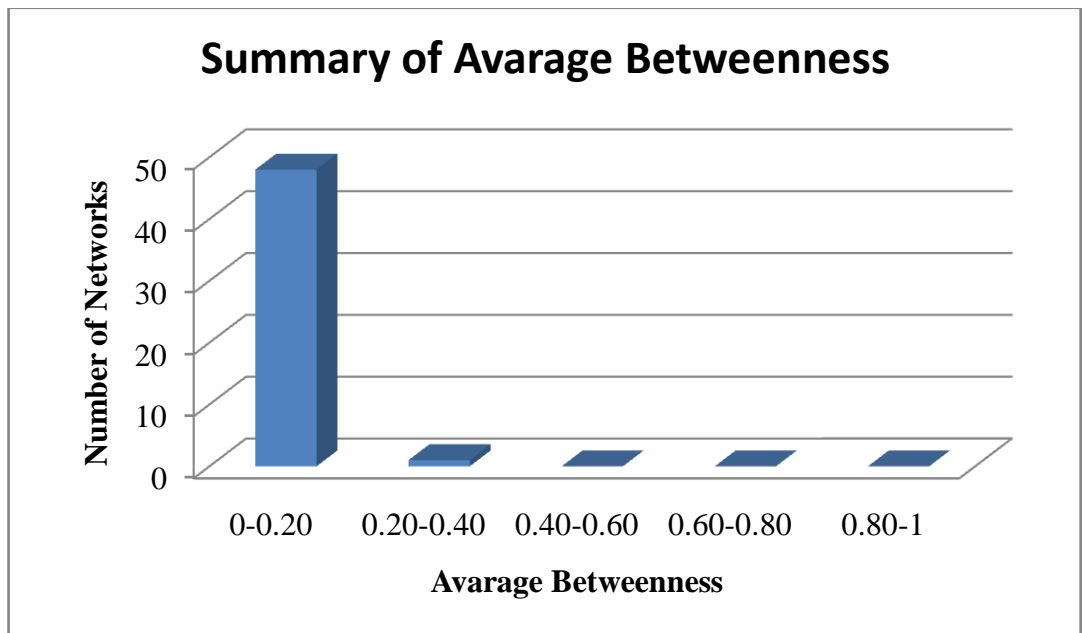


Figure C.5. Summary of avarage betweenness values

REFERENCES

1. Cernea, D.A, D.M. Esther and Jose E. Labra Gayo, "SOAF: Semantic Indexing System Based on Collaborative Tagging", *Interdisciplinary Journal of E-Learning and Learning Objects*, Vol. 4, pp. 137-149, January 2008.
2. Barnes, J. A., "Class and committees in a norwegian island parish", *Human Relations*, no. 7, pp. 39-58, February 1954.
3. Wikipedia Community, *Social Web - Wikipedia, the free encyclopedia*, http://en.wikipedia.org/wiki/Social_web, 2009.
4. Gruber, T., "Collective knowledge systems: Where the Social Web meets the Semantic Web", *Journal of Web Semantics*, Vol. 6, No. 1, pp. 4-13, February 2008.
5. O'Reilly, T., *What is Web 2.0*, <http://oreilly.com/web2/archive/what-is-web-20.html>, 2004.
6. Wunsch-Vincent, S. and G. Vickery, *Participative Web: User-Created Content*, no.DSTI/ICCP/IE(2006)7/FINAL, OECD, 2006.
7. NISO, *Understanding Metadata*, <http://www.niso.org/publications/press/UnderstandingMetadata.pdf>, 2004.
8. Smith, G., *Tagging: people-powered metadata for the social web*, Berkeley, USA, 2007.
9. Huberman, B. A. and Scott A. Golder, "The Structure of Collaborative Tagging Systems", *Journal of Information Science*, Vol. 32, No. 2, pp.198-208, April 2006.
10. Shirky, C., "*Ontology is Overrated: Categories, Links and Tags*", O'Reilly ETech Conference, San Diego, USA, 2005.

11. Mathes, A., *Folksonomies – Cooperative Classification and Communication Through Shared Metadata*, Computer Mediated Communication (LIS590CMC), University of Illinois, Urbana-Champaign, Illinois, 2004.
12. Kim, H.-L., S. Scerri, J. Breslin, S. Decker, H.-G. Kim, “The state of the art in tag ontologies: A semantic model for tagging and folksonomies”, *International Conference on Dublin Core and Metadata Applications*, Berlin, Germany, 2008.
13. Thw, *Twitter – Clone/ Twitter like Sites Collection*, <http://www.thws.cn/articles/twitter-clones.html>, 2007.
14. Heil, B. and M. Piskorsky, "New Twitter Research: Men Follow Men and Nobody Tweets", *Harward Business Review*, June 2009, 2009.
15. Schonfeld, E., *The More Followers You Have, The More You Tweet. Or Is It The Other Way Around?*, <http://techcrunch.com/2009/06/10/the-more-followers-you-have-the-more-you-tweet-or-is-it-the-other-way-around/>, 2009.
16. Cheng, A and M. Evans, *Inside Twitter: An In-depth Look Inside the Twitter World*, Report, Sysomos Inc., 2009.
17. Wettler, M. and R. Rapp, “Computation of Word Associations Based on the Co-Occurrences of Words in Large Corpora”, *International Conference On Computational Linguistics*, Taipei, Taiwan, 2002.
18. Schruz, F.D., *Glossary of Terms Used in Database Searching*, <http://library.iusb.edu/instruction/helpguide/handouts/DatabaseSearching.shtml>, 2009.
19. Hu, Daning and J. Leon Zhao, *Expert Recommendation Via Semantic Social Network*, <http://aisel.aisnet.org/icis2008/196>, 2008

20. Man Au Yeung, Ching, Michael G. Noll, Nicholas Gibbins, Christoph Meinel, and Nigel Shadbolt, *On Measuring Expertise in Collaborative Tagging Systems*. <http://journal.webscience.org/109/>, 18-20 March 2009.
21. Wefollow, *Twitter Directory and Search, Find Twitter Followers: WeFollow*, <http://wefollow.com/>, 2009.
22. Wu, X., Zhang, L., and Y. Yu, "Exploring social annotations for the semantic web", *Proceedings of the 15th international Conference on World Wide Web*, Edinburgh, Scotland, May 23 - 26, 2006, WWW '06, ACM, New York, NY, 417-426, 2006.
23. Mika, P., "Ontologies Are Us: A Unified Model of Social Networks and Semantics", *Journal of Web Semantics*, Vol. 5, No. 1, pp.5-15, 2007.
24. L. Guibault, E.H. Janssen, N.A.N.M. van Eijk, C.J. Angelopoulos, J.V.J. van Hoboken and E. Swart, "User-Created-Content: Supporting a participative Information Society", *Study carried out for the European Commission by IDATE*, TNO and IViR, 2008
25. Halpin, H., V. Robu and H. Shepard, "The Dynamics and Semantics of Collaborative Tagging", *Proceedings of the First Semantic Authoring and Annotation Workshop (SAAW06)*, Vol. 209, 2006.
26. Cattuto, C., C. Schmitz, A. Baldassarri, V.D.P. Servedio, V. Loreto, A. Hotho, M. Grahl and G. Stumme, "Network Properties of Folksonomies", *AI Communications*, Vol. 20, No. 4, pp. 245 – 262, 2007.
27. O'Neill, N., *Twitter Roars Past 14 Million U.S. Users*, <http://www.socialtimes.com/2009/04/twitter-14-million/>, 2009.
28. Alexa the Web Information Company, *Internet Traffic Stats and Metrics*, <http://www.alexa.com/>, 2009.

29. TweetSpeed, *Twitter Instant Speed Meter*, <http://www.tweespeed.com/>, 2009.
30. Wikipedia Community, *Centrality - Wikipedia the Free Encyclopedia*, <http://en.wikipedia.org/wiki/Centrality>, 2009.
31. Linked Data Community, *Linked Data - Connect Distributed Data across the Web*, <http://linkeddata.org/>, 2010.
32. DBpedia, *DBPedia About*, <http://dbpedia.org/About>, 2009.
33. Prud'hommeaux, E. and A. Seaborne, *SPARQL Query Language for RDF*, <http://www.w3.org/TR/rdf-sparql-query/>, 2008.
34. Berners-Lee, T., J. Hendler and O. Lassila, "The Semantic Web", *Scientific American*, Vol. 284, No. 5, pp. 34-43, 2001.
35. C. Cattuto, D. Benz, A. Hotho and G. Stumme, "Semantic Analysis of Tag Similarity Measures in Collaborative Tagging Systems", *3rd Workshop on Ontology Learning and Population OLP3*, Patras, Greece, 22 July 2008.
36. Hotho A., Robert Jäschke, Christoph Schmitz and Gerd Stumme, "Information retrieval in folksonomies: Search and ranking", in York Sure and John Domingue (eds.), Vol. 4011 of LNAI, pp. 411–426, Heidelberg, 2006.
37. Oren E., Sebastian Gerke and Stefan Decker, "Simple Algorithms for Predicate Suggestions Using Similarity and Co-occurrence", *ESWC 2007*, Vol. 4519, pp. 160-174, 2007
38. Twitter Suggested Users List, *Twitter Blog: Suggested Users*, <http://blog.twitter.com/2009/03/suggested-users.html>, 2009.

39. Yamamoto, Y., *Twitter4J - An open-sourced, mavenized and Google App Engine safe Java library for the Twitter API, released under the BSD license*, <http://yusuke.homeip.net/twitter4j/en/index.html>, 2009.
40. TweetStats, *TweetStats :: Graphin' Your Stats*, <http://tweetstats.com/>, 2009.
41. TwitterFacts, *Facts and opinions on twitter and the twitosphere*, <http://twitterfacts.blogspot.com/>, 2009.
42. Sun Microsystems, *Java Technology Reference*, <http://java.sun.com/reference/index.jsp>, 2009.
43. IBM Research, *Eclipse IDE - an open extensible Integrated Development Environment (IDE)*, <http://www.eclipse.org/>, 2009.
44. Batagelj V. and Andrej Mrvar, "Analysis of Large Networks with Pajek", Pajek workshop at XXVI Sunbelt Conference, Vancouver, BC, Canada, 25-30 April, 2006
45. Batagelj, V. and A. Mrvar, "Pajek: Program for large network analysis", *Connections*, Vol. 2, pp. 47–57, 1998.
46. IBM, *MySQL 5.0 Reference Manual*, <http://dev.mysql.com/doc/refman/5.0/en/>, 2009.
47. Twitter, *Twitter API Libraries*, <http://apiwiki.twitter.com/Libraries>, 2009.
48. Twitter, *Twitter API Wiki*, <http://apiwiki.twitter.com/>, 2009.
49. Microsoft, *Microsoft .NET Framework*, <http://www.microsoft.com/.NET/>, 2009.
50. Benslimane D., Schahram Dustdar and Amit Sheth, "Services Mashups: The New Generation of Web Applications", *IEEE Internet Computing*, Vol. 12, No. 5, pp. 13-15, 2008.

51. Merrill, D., *Mashups: The new breed of Web app An introduction to mashups*, <http://www.ibm.com/developerworks/xml/library/x-mashups.html>, 2006.
52. Kamada, T. and S. Kawai, “An Algorithm for Drawing General Undirected Graphs”, *Information Processing Letters*, Vol. 31, No.1, pp. 7-15, 1989.
53. Wikipedia Community, *Microblogging – Wikipedia the Free Encyclopedia*, <http://en.wikipedia.org/wiki/Microblogging>, 2009
54. Wikipedia Community, *Twitter – Wikipedia the Free Encyclopedia*, <http://en.wikipedia.org/wiki/Twitter>, 2009
55. Bebo, *Bebo provides an open, engaging, and fun environment that empowers a new generation to discover, connect and express themselves*, <http://www.bebo.com/>, 2010.
56. MySpace, *MySpace*, <http://www.myspace.com/>, 2010.
57. Flickr, *Flickr Photo Sharing*, <http://www.flickr.com/>, 2010.
58. Del.icio.us, *Social Bookmarking*, <http://delicious.com/>, 2010.
59. Twitter, *Discover what’s happening right now, anywhere in the world*, <http://twitter.com/>, 2010.
60. FriendFeed, *FriendFeed is the easiest way to share online* , <http://friendfeed.com/>, 2010.
61. DeviantArt, *deviantART: where ART meets application!*, <http://www.deviantart.com/>, 2010.
62. Google, *Google Docs*, <http://docs.google.com/>, 2010.

63. Facebook, *Facebook*, <http://www.facebook.com/>, 2010.
64. Google Maps, *Find local businesses, view maps and get driving directions in Google Maps*, <http://maps.google.com/>, 2010.
65. Twitter, *Twitter Search*, <http://search.twitter.com/>, 2010.
66. tinyURL, *TinyURL.com - shorten that long URL into a tiny URL*, <http://tinyurl.com/>, 2010.
67. goo.gl, *Google URL Shortener*, <http://goo.gl/>, 2010.
68. Bit.ly, *Bit.ly, A simple url shortener*, <http://bit.ly/>, 2010.
69. LinkedIn, *Home – LinkedIn*, <http://www.linkedin.com/>, 2010.
70. XING, *Business Network - Social Network for Business Professionals*, <http://www.xing.com/>, 2010.
71. Tumblr, *Tumblelogs are the easiest way to share yourself*, <http://www.tumblr.com/>, 2010.
72. Mashable Social Media Guide, *Mashable | The Social Media Guide Feed*, <http://mashable.com/>, 2010.
73. ReadWriteWeb, *ReadWriteWeb - Web Apps, Web Technology Trends, Social Networking and Social Media*, <http://www.readwriteweb.com/>, 2010.
74. YahooTravel, *Yahoo! Travel - Airline tickets, cheap hotels, cruises, vacations & honeymoon travel*, <http://travel.yahoo.com/>, 2010.
75. Trip Advisor, *Hotels and Vacation Reviews – TripAdvisor*, <http://www.tripadvisor.com/>, 2010.

76. IMDB Community, *Internet Movie Database*, <http://www.imdb.com/>, 2010.
77. Zarella, D., *Modeling ReTweet Dynamics*, <http://danzarella.com/modeling-retweet-dynamics.html>, 2009.
78. Wikipedia Community, *Tag (Metadata) - Wikipedia, the free encyclopedia*, [http://en.wikipedia.org/wiki/Tag_\(metadata\)](http://en.wikipedia.org/wiki/Tag_(metadata)), 2009.
79. Wikipedia Community, *Metadata – Wikipedia, the free encyclopedia*, <http://en.wikipedia.org/wiki/Metadata>, 2009
80. Wikipedia-Community, *User-Generated Content - Wikipedia, the free encyclopedia*, http://en.wikipedia.org/wiki/User-generated_content, 2009.
81. Wikipedia-Community, *Application Programming Interface - Wikipedia, the free encyclopedia*, http://en.wikipedia.org/wiki/Application_programming_interface, 2009.
82. Wikipedia Community, *Wikipedia – The Free Encyclopedia*, <http://wikipedia.com/>, 2010.
83. Google, *Google Search Engine*, <http://www.google.com/>, 2010.
84. DBPedia Community, *DBPedia*, <http://dbpedia.org/About>, 2010.
85. The Computer Language Company, *Computer Desktop Encyclopedia*, http://lookup.computerlanguage.com/host_app/search, 2010.
86. Wikipedia-Community, *HTTP*, http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol, 2010.
87. Wikipedia-Community, *XML*, <http://en.wikipedia.org/wiki/XML>, 2010.

88. Wikipedia-Community, *JSON*, <http://en.wikipedia.org/wiki/Json>, 2010
89. Twitterholic, *Top Twitter User Rankings & Stats*, <http://twitterholic.com>, 2010.