Mentor: Sourav Kr. Dandapat

1. Exploiting heterogeneity in WLAN

According to the capabilities of wireless devices they are categorized as

Type1:Bluetooth enabledType 2: WiFi enabled Type 3: both Bluetooth and WiFi enabled.

Our objective is to build a prototype where different kind of devices can interact among themselves. If two devices of same type want to communicate, then it becomes simple; otherwise it needs conversion. Suppose a Type1 device wants to communicate to Type3device. In path of this communication, it must go through a Type2 device. Type2 device receives a Bluetooth communication, convert the format according to WiFi and sends it to Type3 device.

2. Localization in wireless network

Localization is a method of finding location. For wireless devices, some time it becomes very important to know its location. It may be helpful to reach a destination, hugely used in sensor network to cooperate and collect data. In this project, we want to compare different localization methods (at least 4 methods: including both indoor and outdoor localization method) through simulation. Study their power requirement and limitations.

3. Distributed trust and incentive scheme for collaborative network.

For content distribution in wireless network, collaboration is very important. One important question is to whom one should collaborate? Due to the unavailability of central server some distributed techniques become necessary in many situations such as to fight against attack and security breach.

In this project, we want to study several existing trust and incentive mechanism and want to propose a new trust and incentive scheme.

Mentor: Saptarshi Ghosh

4. Studying transportation networks in India from a Complex Network Perspective

The structure and efficiency of transportation networks are of primary importance in the economic development of any country. In this term project, we shall analyze the structure of the Indian Railway network and the network of highway-roads in India, using tools of Complex Network Theory. This will involve collection of relevant data and comparative analysis of the two networks.

5. Studying various node attachment policies in bipartite networks

Several real-world systems can be modeled and analyzed as bipartite networks where an active set of entities selectively associate with a passive set of entities; some examples are

- membership of users in groups in Online Social Networks (e.g. Facebook groups),where the users (active entities) become members of groups (passive entities)
- railway systems, where trains (active entities) stop at some selected stations (passive entities)

We model such systems as: when active entity x associates with passive entity y, we put an edge (x,y) in the bipartite graph. In such systems, an interesting and important aspect is: how do the active nodes decide which passive node to attach with (i.e. create edges with)? In this project, we shall study different mechanisms of node attachment policies through simulation. We shall also see which policy best matches with various empirical data (e.g. we have user-group membership data of several Online Social Networks, train-station data of Indian Railway Network, etc).

6. Data-Collection and Analysis on Online Folksonomies

Several online social systems (e.g. Flickr, Delicious) are folksonomies where 'users' annotate 'resources' with 'tags' e.g. a user in Flickr can tag a photo of Sachin Tendulkar with the words 'Tendulkar', 'cricket', 'batsman', etc. Such collaborative tagging helps in automatic classification of resources and helps in searching for the resources e.g. when a user is searching for an image of Sachin Tendulkar, a good answer for his query would be a photo which many other users have tagged with 'Tendulkar'.

In this project, we aim to analyze the structure of online folksonomies (the term folksonomy is explained in the project above). This will involve designing a crawler for large-scale collection of data from the Flickr website (a very popular photo-sharing folksonomy) and analysing the topological properties of the network. Knowledge of Python programming language will be useful for this project, since Flickr gives a good Python API for crawling.

7. READING PROJECT: Community Detection in Folksonomies

A community in a network refers to a group of 'similar' nodes, where similarity is defined by different people in different ways, but with the underlying assumption that similar nodes with have more edges among themselves and hence will form a densely-connected sub-graph in the network. For instance, close friends are expected to form a community in an online social network like Facebook.

Finding communities in folksonomies (this term is explained in the project above) in an interesting and challenging problem which is currently under active research. In this reading project, we shall study and compare different methods proposed to find communites in folksonomies, and the various use of such methods (e.g. in recommending friends and resources to users).

Mentor: Parantapa Bhattacharya

Introduction

For all of the following networks generate the graph. Observe the degree distribution, clustering coefficients, spectral phenomena, and other network properties for the networks. Check for cycles in the graph. If they exist explain their occurrence and effects.

8. Header Files Network

C header files include other header files. Create a digraph with header files as vertices. There exists a directed edge A to B if header file A includes header file B. Generate the header file network for all header files in the /usr/include directory. The program written for extracting the network should work on any folder with header files in them.

9. Shared Libraries Network

Shared libraries in GNU+Linux systems (.so files) can use or link to other shared libraries. Create a digraph with shared library files as vertices. There exists a directed edge from A to B if the shared library A links to shared library B. Generate the shared libraries network for all shared libraries in /lib /lib32and /lib64 directories.

```
[parantapa@osiris: ta-duty]$ ldd /lib/libglib-2.0.so.0
linux-vdso.so.1 => (0x00007fffedbff000)
libpcre.so.3 => /lib/libpcre.so.3 (0x00007f00b45a0000)
libc.so.6 => /lib/libc.so.6 (0x00007f00b423f000)
/lib64/ld-linux-x86-64.so.2 (0x00007f00b4ac7000)
```

10. Package Dependency Network

Package management softwares such as apt, yum use an internal database to manage dependencies among software packages. Create a digraph with packages as vertices. There exists a directed edge form A to B if package A dependson package B. Generate the package dependency network for all packages in the Debian Lenny main, contrib, and non-free repository and Fedora 12fedora and fedora-updates repository. The aptitude command in Debian and yum in Fedora can show package information which includes dependencies. But for the project you need to download the whole repository metadata for program input.

```
[parantapa@osiris: ta-duty]$ aptitude show build-essential
Package: build-essential
...
Version: 11.5
...
Depends: libc6-dev | libc-dev, gcc (>= 4:4.4.3), g++ (>= 4:4.4.3), make,
dpkg-dev (>= 1.13.5)
Description: Informational list of build-essential packages
...
```

11. Function Call Graph

Create the function call graph for the Linux 2.6.37 kernel source code and the gcc 4.6 source code. Use each function as a vertex. There exists a directed edge between function A and B if function A calls function B. Use a name mangling scheme for functions with same name in different files.

Mentor: Rishiraj Saha Roy

QUERY LOG RELATED ASSIGNMENTS

12. A study of word co-occurrence networks from natural language text.

Complex networks can be found everywhere, even among the words in natural language corpora. In this assignment, you will study the properties of word co-occurrence networks

(*NWCN*, *N* denotes natural language). Every unique word is a node in this network, and two nodes share an edge if they co-occur in a sentence in a document. Edge weights are defined to be the number of times of co-occurrence. We wish to observe the degree distribution, clustering coefficients, spectral phenomena and other network properties of the *NWCN*. We will be using the EuroParl English corpus for our experiments.

13. A study of word co-occurrence networks from query logs.
The *WCN* is defined in a similar fashion as in the previous assignment, but in this case a document consists of a single query. We call it a *QWCN*. Every unique word from a large Web search query log is a node. There is an edge between two nodes if they co-occur in a query, with the edge weight being the number of queries they co-occur in. We compute the same properties for this network as for the *NWCN*. The goal is to compare the properties of the *NWCN* and the *QWCN* and make interesting inferences. We will use the AOL query log as our experimental dataset.

NOTE: Groups for Term Projects 12 and 13 will work in close collaboration.

14. A comparison of the query network as obtained by projections from two different bipartite networks.
In our Web search query logs, we have queries, their clicked URLs, and the corresponding click counts. We can define a *QW* (Query-Word)-bipartite network that have all unique queries in one partition and all unique words in the other partition. An edge exists between a query and a word if the word is contained in the query. The edge weight is the number of times the word occurs in the query, and hence, is mostly one in value. We can obtain a projection of this *QW*-network using a 2-steprandom walk, yielding a *QQ*-network. Similarly, we can define a *QU* (Query-URL)-bipartite network that has all unique queries in one partition and all unique URLs in the other. An edge exists between the two partitions, i.e. between a query and a URL, if the URL is clicked for the query. Edge weights are the respective click counts. A projection of this *QU*-network will also give us a *QQ*-network. We wish to observe interesting similarities and differences in the *QQ*-network properties as obtained from the two independent projections. We will use the AOL query log as our experimental dataset.

15. A comparison of the clustering of the query network as obtained by projections from two different bipartite networks.
This task overlaps partially with the previous term project. After we obtain the *QQ*-networks by the two methods, we implement a graph clustering scheme to cluster the respective networks. The challenge is to gain insights from the comparison of the two sets of clusters.

NOTE: Groups for Term Projects 14 and 15 will work in close collaboration.

16. Reading assignment. A review of graph-based techniques in natural language parsing.
Graph-based techniques seem to hold great promise in discovering a structure in queries, which till now, are largely considered to be bags-of-words. However, such techniques

have been successfully applied in the field of natural language processing. Examples include techniques to build projectivity trees, dependency trees and constituency trees for natural language grammars. The task in this reading assignment is to perform a comprehensive study of the past work done in this field. More specifically, you have to summarize papers which have used graph-based techniques in natural language parsing, and identify if there exist works that have used them in unravelling a structure in queries.

Mentor: Joydeep Chandra

17. Study of the Online News Article Network

For a set of online news article, we can consider two articles as related if they share some tags. Crawl any news website like www.timesofindia.com. Create a network of news article nodes, where an edge will exist between two news articles nodes if they share one or more tags. The number of tags they share forms the weight of that link. Further associate a strength value to each of the nodes, where the strength value is a measure of popularity that can be considered as the number of comments it has received. Study the various properties like degree and strength distribution, weighted clustering coefficient, assortativity, centrality, modularity etc. and try to comment on the insights obtained from each of these measures.

Reference:
A. Barrat, M. Barthélemy, R. Pastor-Satorras, and A. Vespignani, The architecture of complex weighted networks, PNAS 2004 101: 3747-3752.

18. Network of Bloggers

Build a network of bloggers, who comment on the news articles published in leading online newspapers based on their ID and location. Consider 2 bloggers as connected if they have posted their comments on the same news article. The weight of the link between two bloggers is determined by the number of news articles in which both has commented.

1. Build the network and study the properties of the weighted network mentioned in the reference network and try to draw interesting insights.
2. Since the news articles contain the time at which a blog has been posted, study the growth of the blog network. Further try to comment about the growth of the network observed for the popular news articles for various categories like
politics, sports, entertainment, lifestyles etc.
3. Collect the bloggers information for Indian, Chinese and US newspapers and compare the various properties of the networks and state the insights that can be obtained.

Reference:
A. Barrat, M. Barthélemy, R. Pastor-Satorras, and A. Vespignani, The architecture of complex weighted networks, PNAS 2004 101: 3747-3752.

19. The Amazon Referral Network

Crawl the books.amazon.com website. In every book information page you will find a link that states “Customers Who Bought This Item Also Bought”, followed by the links

to the books. Build a network of these books, where an edge between two books exists if one of them is liked by the reader of the other.
1. Study the properties of the network thus formed and state the interesting insights drawn from the study.

2. Further crawl for suitable information that might also answer these questions like,
a. Will a book likely to be popular if any one of its author is popular?
b. Books co-authored by popular authors are likely to be more popular than books written by a single popular author?
c. Discuss the assortativity properties of the co-author networks measured in terms of the author's popularity.
Note: *One measure of popularity can be the weighted average of the ratings obtained from the reviewers.*

Reference: M.E.J. Newman, The Structure and Function of Complex Networks. SIAM Review 45, 167-256 (2003)

20. Survey: Survey on Dynamic Network Analysis.
Do an elaborate survey on Dynamic Network Analysis. You should be able to identify the problem areas that require DNA, the challenges involved and why traditional social network analysis fails in these situations, the identified problem areas and the issues being worked upon in this area and the solutions proposed. Remember you have to prepare a wiki page and also an elaborate report of around 20 pages on this topic.

Reference:
http://www.chronicdisease.org/files/public/2009Institute_NA_Track_Carley_2003_dynamicnetwork.pdf

Mentor: Rajib Maiti

21. Comparative analysis of performance of SIRS epidemics for different mobility models:
Information spreading in DTN using OA(omni-directional antenna) and DA(directional antenna) where mobility model is RW and group with a single source (broadcasting in particular).
-- comparative analysis of the improvements or degradation of results of OA and DA.

22. Modeling Human daily life mobility pattern and information spreading on such mobility model using SIRS epidemics:
Generating human mobility pattern (random mobility pattern, super preferential mobility pattern, group mobility pattern) : given a set of popular places and a number of agents. referring paper of SLAW(self-similar least action walk).

Mentor: Animesh Srivastava

23. Analyzing the impact of edge removal attack (bond percolation) on the correlated networks.

24. Analyzing the impact of node removal strategies on the assortativity of the real-world networks e.g. Internet, Facebook etc.

25. Studying the spectral graph properties of real-world correlated networks e.g. Twitter and Facebook.

Mentor: Sudipto Saha

Experiments on random walks in complex networks

Efficient decentralized search/dissemination with minimum overhead and overlap - has been one of major issues of research in the recent past. The existing algorithms are either based on flooding or random walk. In one extreme side, flooding based algorithms are there. They can be thought of as a deterministic proliferating algorithm, where the start nodes initiates the process with a single walker - which proliferates in maximum possible way/number in the following steps - i.e. - it generates walkers for all the paths available from the start node. On the other end of the spectrum there are the algorithms based on simple single random walk. It can be thought of as a nondeterministic zero-proliferation algorithm - which chooses only one of the available paths at a time. There are different types of search/dissemination algorithms – which works in between these two extreme ends. We can think of a proliferated and biased random walk - in which, when a walker comes at a certain node - having a certain degree, it can be proliferated and/or forwarded to certain fraction of its neighbors with a certain probability. This probability should be determined locally and should reversely vary with the probability that the neighbor has already been visited by a random walker at that time step. Now, determining this probability and the particular proliferation rate (based on purely local information) which minimizes the interferences among the walkers and also maximizes the number of distinctly visited nodes densely around the start node – is a big challenge. We can very crudely approximate that probability with the ‘first passage probability’ of visiting of the node by some random walker at a particular time step.

Based on the above facts we have three projects for three groups –

26. Do a thorough literature survey on the existing work on biased / proliferated random walk – on different types of networks – including peer to peer networks - and extract the features of the existing algorithm, what is the maximum coverage it can achieve within a particular time and consuming provided bandwidth, comparisons of their different features.
Develop some variation of biasing and proliferation techniques which can give better performance in all these networks within the provided constraints. [One variation you can try is - randomly biased random walk]

27. The networks which can typically be simulated by 2-D Euclidian grid, plain random walk techniques waste too much bandwidth – as in 2D the random walk is persistent for which, it returns to the origin – infinitely often. Therefore, first do a thorough survey of the works done on the random walk (necessarily biased) based in those networks. After that verify the applicability of the randomly biased random walk with optimal proliferation techniques in this area and compare its performance.

28. Develop an efficient and expandable Graphical User Interface (GUI) which will be able do at least the following functionality now –

a. Generation of the Euclidian grids of 1D, 2D, and 3D with any number of node and width. [Also try to - generate random graphs with given degree distribution (both from parameters and from file – like pajek)]

b. Display of the dynamic processes like – single random walk, K-random walk, biased random walk, proliferated random walk – on these graphs – and show the results.