A Survey of Social Software Engineering

Francesco Lelli*

University of Lugano

* On behalf of Navid Ahmadi, Mehdi Jazayeri, Sasa Nesic.

Outline

- Introduction
- Software Engineering as Social Activity
- On Engineering of Social Applications
- Mathematical Methods for Social Software Engineering
- Conclusions

Why People in Software Engineering?

Today software development is:

- *Carried out in teams*
- *These teams often include domain specific experts*
- *Most of the time is spent in understanding the problem, the division of the work among the members, and in understanding the requirements that a particular component should meet.*

A Few Numbers

- On large projects typical systems developers spend about 70% of their time working with others. [1]
- Team activities account for about 85% of the costs of large software systems. [2]

[1] DeMarco, T., & Lister, T. (1987). *Peopleware:productive projects and teams. New York, NY, USA: Dorset* House
[2] Publishing Co., IncJones, C. (1986). *Programming productivity. New York, NY, USA: McGraw-Hill, Inc.*

Software Engineering as Social Activity

Useful input could come from:

- Empower People (Weinberg 1971)
- Human Centric Software Development.
- Psychology of Small Groups
- Working as a Social Activity.
- Psychology of Persuasion
- Persuasive Computing

Collaborative Applications

- Need for supporting social aspects by means of collaborative tools
- Computer Supported Cooperative Work (CSCW) studies collaborative application. But Software Engineering is collaboration finalized in building Artifact.
- Collaborative Development Environments: the collaborative features are integrated into the development environment.
- Impact of the Web and Web 2.0 on software engineering

Web-based Collaboration on Software

- Web as an enabling technology
  - It has changed the software development providing a cheap medium for large-scale software distribution, update, and reuse. Based on Hypermedia and URI.
- Collaborative Sites
  - Basic collaboration using Web
  - Common feature categories: co-ordination, communication and community building
  - Proposed Layers: Project workspaces, team tools, development resources

Web 2.0 Fosters Software Engineering

- Web 2.0: leveraging end-user participation to building collective intelligence
  - Wikis: a concrete Web 2.0 example
- Wikis as document repository in software engineering
  - Share and reuse of documents
  - Weaving documents in heterogeneous environments
  - Capturing architectural design rationale
- Programmable Wikis
  - Wikis as programming Environment
  - Inherently a collaborative environment
  - Executable pages: pages are equal to functions

Social Network Services (SNSs)

- SNSs are promoted as central to Web 2.0;
- Common functionalities:
  - a personal profile page, a network of friends listings, private messaging, discussion forums, events managements, and commenting and media uploading;

Well known problems:

- Diversity of SNSs hampers the interoperability;
- There is no shared understanding of objects' meaning;
- Relationships among users as well as shared objects are represented only for human consumption;

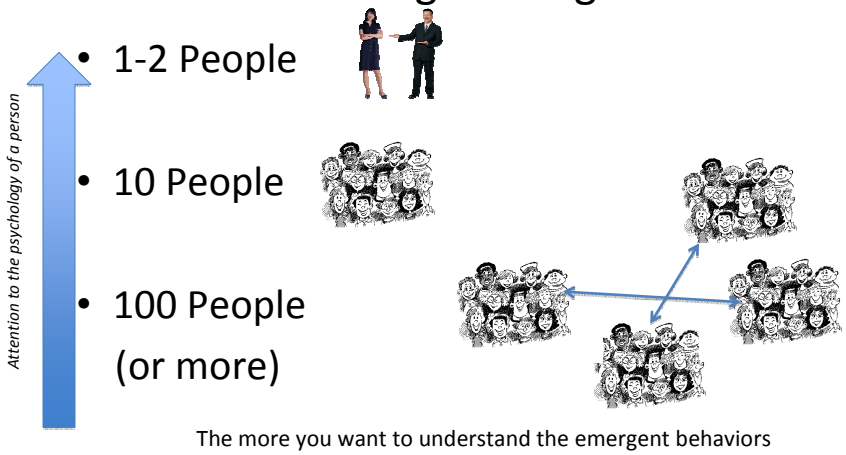
Leveraging the Semantic Web technologies for SNSs

- Semantic Web technologies:
  - Universally unique resource identification (URIs);
  - Common formats for resource descriptions (RDF);
  - Agreed models for domain conceptualization (Ontologies);
  - Formally specified ontology languages (RDF-S, OWL);
  - RDF query language and data access protocol (SPARQL);
- Combining SNSs and the SW offer potential benefit;

Benefits for SNSs

- Use of URIs within SNSs enables identification of users and shared objects independently of particular SNS;
- By publishing users' and shared object's descriptions as RDF, the SNSs enable them to be shared with other SNSs;
- Use of ontologies enables understanding of the meaning of those descriptions;
- Existing initiatives for creating universally recognized ontologies:
  - FOAF (Friend-of-a-Friend);
  - SIOC (Semantically Interlinked Online Communities);
  - PIMO (Personal Information Management Ontology);

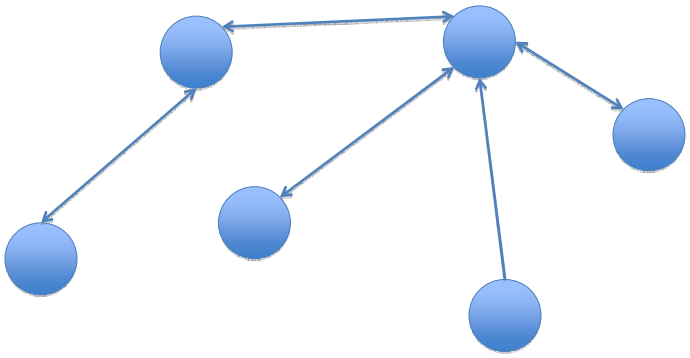
Mathematical Methods for Social Software Engineering



The more you want to understand the emergent behaviors of a large group of people, the less important are the attributes of each single person

Social Network Analysis

- A social network is a set of people in a relation



Social Network Representation

Each network link is represented by a random variable (often binary)

For i,j members of some set of actors N

$X_{ij} = 1$ if there is a network link from person i to person j

$X_{ij} = 0$ if there is no link

Directed network: $X_{ij} \neq X_{ji}$

Non Directed network: $X_{ij} = X_{ji}$

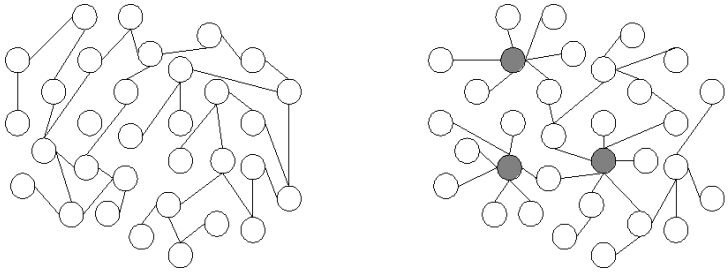
X is a matrix of all nodes

Probability of $X_{ij} = 1$

Uniform Distribution: $P[X \geq x] \approx cx^{-\alpha}$

Power law distribution: $P[X_{ij} = 1] \approx \alpha$

What does it mean?



(a) Random network (b) Scale-free network

In scale power law based network (or free network) the highest-degree nodes are often called "hubs", and are thought to serve specific purposes in their networks, although this depends greatly on the domain.

A Few Metrics in Social Networks

- The **degree** of a node is the number of ties incident to the node.
- The **closeness** of a node represents how much an individual is near to other individuals in a network (directly or indirectly).
- The **structural cohesion** is the minimum number of members who, if removed from a group, would disconnect the group.
- The **path length** represents the distance between pairs of nodes in the network.
- Many more…

Formalization of the Prisoner Dilemma

	Prisoner B Stays Silent	Prisoner B Betrays
Prisoner A Stays Silent	Each serves 6 months	Prisoner A: 10 years Prisoner B: goes free
Prisoner A Betrays	Prisoner A: goes free Prisoner B: 10 years	Each serves 5 years

Some considerations about the Dilemma

- The prisoners do not have a global view.
- The local optimum is different than the global optimum.
- Collaboration is a win-win situation.
- But…It involves trust.
- A Prisoner, even if he understands the game may not make the "right move"
- Do you really believe that you should not care about the mindset of the prisoners?

Conclusion

1. We outlined the need to integrate results from social and psychological sciences in the software lifecycle;
2. We outlined the need for engineering social networking services and collaborative tools.
3. We presented a set of mathematical techniques that may be used as basic blocks for understanding the ideas proposed for social software engineering.

Question?

- Actually… I have one…

mmm…… Social Software Engineering……
maybe…

Question?

- Thx for your time!