Mining Social Topologies from Email for Online Data Sharing

Diana MacLean, Sudheendra Hangal, Seng Keat Teh, Monica Lam and Jeffrey Heer

Stanford University
Introduction

Algorithm

Social Flows Interface

Evaluation

Conclusions
A Social Topology
Social Topologies

The Definition:
“the structure and content of a person’s social affiliations, consisting of a set of overlapping social groups and the subset/superset relationships between them.”

The important bits:
- overlapping
- nested
- extremely granular
A Snapshot of Today

- Online experience increasingly *social*.
- Social ties handled at each point of contact.
  - Facebook friends’ lists
  - Gmail contact groups
  - Dropbox communities ...
So what’s wrong with this picture?

- Requires *manual* setup (at each point of contact)
- Requires *manual* maintenance (at each point of contact)
- Does not scale
Our Vision

Your Social Topology is captured *latently* in your daily communication patterns anyway (think: e-mail).

- Mine it
- Maintain it
- Port it (or parts of it) to online services
Why e-mail?

- Everyone has it
- Spans several years
- Good labeled data
- Reflects changes in relationships over time
Let every unique recipient set be a group.

- too many groups
- incomplete
- lacking macrostructure
- it’s all about pruning
Our Attempt: Properties

creates a social topology from a sent mail folder that is

▷ ~80% smaller than naive model
▷ ~95% smaller than naive model with collapsed hierarchy
▷ easily navigable
▷ accurate and complete (more about this later)
Our Attempt: Outline

4 components:

- Phase 0: Data preparation/cleaning
- Phase 1: Extracting “social molecules”
- Phase 2: Merging “social molecules” into larger groups
- Phase 3: Organizing results into a hierarchy
Phase 1: Extracting “Social Molecules”

“A social molecule is a small group of people that comprise a relevant, logical social unit according to the users communication patterns.”

Obvious proxy: unique, frequent recipient sets

Important property: individuals may belong to several social molecules
Phase 1: Extracting “Social Molecules”

Let $S$ be the set of social molecules.

Add each unique message recipient set $s$ to $S$ if $s$ has high enough frequency in the corpus.
Phase 1: Extracting “Social Molecules”

Let $S$ be the set of social molecules.

Add each unique message recipient set $s$ to $S$ if $s$ has high enough frequency in the corpus.
Phase 1: Extracting “Social Molecules”

\[ S = \{ \]
\[
\text{Mom, Dad, Grandma} \quad 10 \\
\text{Mom, Dad, Sister, Brother} \quad 30 \\
\text{Sister, Brother} \quad 62 (32) \\
\text{Mom, Dad, Sister} \quad 31 (1) \\
\text{Sister, Grandma} \quad 9 (4) \\
\text{Sister, Brother, Grandma} \quad 5 \\
\}

Add to \( S \) all the pairwise intersections of \( S \), under the same criteria.
Phase 1: Extracting “Social Molecules”

\[ S = \{ \text{Mom, Dad, Grandma}^{10}, \text{Sister, Brother}^{62 (32)}, \text{Mom, Dad}^{40}, \text{Mom, Dad, Sister}^{31 (1)}, \text{Mom, Dad, Brother}^{30}, \text{Sister, Grandpa}^{9 (4)}, \text{Sister, Brother, Grandma}^{5} \} \]

Add to \( S \) all the pairwise intersections of \( S \), under the same criteria.
Phase 1: Extracting “Social Molecules”

\[ S = \{ \text{Mom, Dad, Grandma} : 10, \text{Sister, Brother} : 62 (32), \text{Mom, Dad} : 40, \text{Mom, Dad, Sister} : 31 (1), \text{Sister, Grandma} : 9 (4), \text{Sister, Brother, Grandma} : 5 \} \]

Retain only those \( s \) with sufficient self identity

- The sharing error between \( s_i \) and \( s_j \), \( s_j \subset s_i \) is a measure of information leaked if \( s_i \) and \( s_j \) were merged.

\[ serr(s_i, s_j) = \frac{|s_i| - |s_j| \times (msgs(s_j) - msgs(s_i))}{|s_i| \times msgs(s_j)} \]

- If \( serr(s_i, s_j) \) is large, we retain \( s_j \), otherwise we say \( s_i \) subsumes \( s_j \).
Phase 1: Extracting “Social Molecules”

S =

Mom, Dad, Grandma
10

Sister, Brother
62 (32)

Mom, Dad
40

Mom, Dad, Sister, Brother
30

Mom, Dad, Sister
31 (1)

Sister, Grandma
9 (4)

Sister, Brother, Grandma
5

Retain only those s with sufficient self identity

- The sharing error between \( s_i \) and \( s_j \), \( s_j \subset s_i \) is a measure of information leaked if \( s_i \) and \( s_j \) were merged.

\[
serr(s_i, s_j) = \frac{|s_i| - |s_j| \times (msgs(s_j) - msgs(s_i))}{|s_i| \times msgs(s_j)}
\]

- If \( serr(s_i, s_j) \) is large, we retain \( s_j \), otherwise we say \( s_i \) subsumes \( s_j \)
Phase 2: Merging Social Molecules

But we’re still missing macro-structure.

For each $s_i$, $s_j$, add $(s_i \cup s_j)$ to $S$ if $s_i$ and $s_j$ are sufficiently "similar". Note that:

- use Jaccard similarity according to set membership with some threshold
- note: no sets are discarded
Phase 2: Merging Social Molecules

But we’re still missing macro-structure.

For each $s_i$, $s_j$, add $(s_i \cup s_j)$ to $S$ if $s_i$ and $s_j$ are sufficiently “similar”. Note that:

- use Jaccard similarity according to set membership with some threshold
- note: no sets are discarded
Phase 3: Organizing $S$ into a Hierarchy

- Find $s'$, the $s$ with greatest group mass in $S$. $s'$ is a parent group
  - group mass of $s = \text{sum of msgs attributed to each person in } s$
- Assign all $s$ similar to $s'$ as children groups
  - Again, use Jaccard similarity according to set membership with some threshold
Applications to other data...

 Applies to almost any data with group tagging, and in which frequency $\sim$ importance.
Our interface allows easy browsing and manipulation of social topologies.

Annotated points of interest highlight: (a) hierarchical nesting of subsets; (b) editable group labels; (c) tooltip and delete option on mouse hover; (d) new group creation; (e) group merge tools; (f) additional group editing tools and (g) option to add a new contact.
Using a.) Gmail Contacts tool, and b.) Social Flows, ask users to create partial social topologies congruent to contrived scenarios.
6 out of 19 users found the Gmail interface intolerable and quit the task!
User Study Evaluation: Results

- Topology creation in Social Flows is significantly faster
- Social Flows interface is significantly easier to use
- Resulting topologies in Social Flows are significantly more satisfactory
- Resulting topologies in Social Flows are more useful for online sharing tasks.
User Study Evaluation: Conclusions

- Our algorithmically generated templates:
  - reduce overhead construction time of social topologies
  - reduce cognitive recall required to remember group membership
  - are reasonably accurate

- Our Social Flows interface is an improvement over state of the art in managing social contacts
Take Aways

- Social Topologies data structure
- Algorithm to generate overlapping social groups (first that we know of)
- Port to sites for centralized sharing/access control
Introduction

Algorithm

Social Flows Interface

Evaluation

Conclusions