Workshop Proceedings of the 9th International Conference on Intelligent Environments
J.A. Botía and D. Charitos (Eds.)

doi:10.3233/978-1-61499-286-8-90

Social sub-group identification using social graph and semantic analysis

Bjørn Magnus MATHISEN [a,1], Anders KOFOD-PETERSEN [a], John MCGOVERN [b], Thomas VILARINHO [a] and Babak A. FARSHCHIAN [a]

[a] *SINTEF ICT, S. P. Andersens vei 15 b, 7465 Trondheim, Norway*
[b] *TSSG, ArcLabs Research & Innovation Building,WIT, West Campus, Carriganore, Co. Waterford, Ireland*

Abstract.
Social networks proliferate daily life, many are part of big groups within social networks, many of these groups contain people unknown to you, but with whom you share interests. Some of these shared interest are based on context which has real-time properties (e.g. Who would like to go to a jazz concert during an AI conference). This paper presents a possible method for identifying such subgroups. We aim to do this by creating a social graph based on one or more "comparators"; Frequency (how often two members interact) and content (how often the two talk about the same thing). After the graph is created we apply standard graph segmentation (clique estimation) techniques to identify the subgroups. We propose a system which continuously polls social network for updates, to keep the social graph up-to-date and based on that suggest new subgroups. As a result the system will suggest new subgroups in a timely and near real-time manner. This paper will focus on the methods behind the creation of the weighted social graph and the subsequent pruning of the social graph.

Keywords. Social Networks, Social graph, graph analysis, semantic analysis, text mining.

Introduction

In recent years, social networks have become prevalent throughout society [1,2]. The increasing importance of social networks drives the increase of users and traffic. However, increasing the number of users and traffic can easily present itself as a problem to most users. The "noise", that is irrelevant information, increases and it becomes harder to discriminate which users and traffic to tune in to, which groups to join and which users you want to interact with. Thus, developing tools that can help to identify subgroups that are more relevant to a user, or a group of user is becoming more important.

Recently, work has been done to uncover methods for identifying community structures from the textual social interaction within a existing structure, see e.g. [3,4]. Most of these works is centred around the analysis of the social graph, where vertices (people) are joined together by links or edges (communication) [5]); different clustering

[1] Corresponding Author: Bjørn Magnus Mathisen, SINTEF ICT, S. P. Andersens vei 15 b, 7465 Trondheim, Norway; E-mail: bjornmagnus.mathisen@sintef.no.

techniques largely developed from different metrics of graphs [6,7,8,9,10], with some of them having slightly different perspectives [11,12].

The work presented here is part of the on going Societies EU project[2] and demonstrates how directed and broadcast messages in social networks can be harvested and parsed to build new social graphs. The main focus of this paper is the algorithm that constructs the social graphs. The rest of the paper is organised as follows: Section 1, give an overview of relevant related work; Section 2 puts the graph building algorithm into context by briefly describe the overall architecture; this is followed by a description of the algorithm developed. The paper ends with a summary and outlook on future work.

1. Background and Related Work

Investigating structures and communication patterns in different forms of social networks has in the last decade received a growing interest.

Social graph are typically constructed by assigning people to the vertices and their interaction to the edges. This social interaction can be many things, but traditionally strength of connections have been popular. The strength is typically measured by counting number of messages. One such example is, Tyler et al., who describe how communities of practice can be identified from email correspondence within organisations [13]. The communities are identified solely from the sender and receiver of the email correspondence. The process was a two step process for constructing a graph. The graph consisted of nodes that were the senders and receivers of emails and links emails that connected the persons. The graph was then used to identify sub-graphs, or communities of nodes with many links between them.

The idea of counting the frequency of communication can also be employed in more loose settings. As an example, Gómez et al., build a social network based on the implicit relationship between authors of comments by other users on Slashdot.org [14]. The graphs build here does not represent a classic social network, but rather dynamic loose networks based on shared interests.

Assuming that interests form social networks, it would not by unreasonable to assume that these social networks would also share a lingo. Bryden et al, does indeed demonstrate that communities based on frequency of messages is closely mirrored by communities based on frequency of words [15]; that is people share a vocabulary with the people they communicate with.

Communication sharing the same lingo and interests have also been used to build social graphs. Examples are Lui et al., [16] and Anwar and Abulaish [17]. The latter further extends the notion of communities and language. They apply text mining techniques to generate social graphs based on similar posts. The authors crawls internet forums and maps, similar to [14], a *reply-to* relationship. Yet, they extend this by also clustering messages based on their similarity. This results in a social graph where the edges are a sum of the *reply-to*, that is message frequency and similarity.

[2]Project number: ICT-257493

2. System

The social graph builder deals with building social graphs centred around relevant users and groups.

The system is based on analysing activities (a subset of activities as defined in [18]), which are modeled as data-types consisting of triplets on the form: $[sender, receiver, message]$. The set of activities harvested is parsed into subset grouped by the sender. Such a subset will contain all activities and textual output from that sender, both directed (e.g. IM message) and broadcast (e.g. a tweet). All unique permutations of possible pairs of these subsets are then fed into the comparators. These comparators calculate a weight for the link between the two nodes in the social graph, if the two nodes already have a connection the weight is added onto the existing weight. This algorithm is detailed in section 3.1.

Figure 1 depicts the general system architecture which is composed of comparators acting as components in an ensemble system which rates the strength of a edge in the social graph.

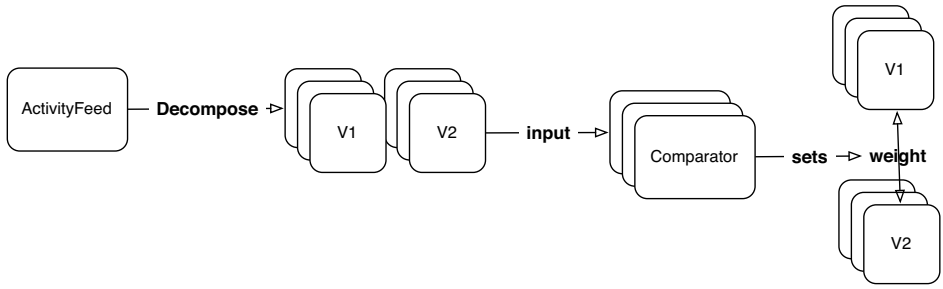

Figure 1. The system architecture

Figure 2 depicts the functional implementation of the architecture, as of now we are running three comparators:

1. **Frequency**: This comparator calculates a weight on the edge based on the frequency of communication between the two social graph vertices.
2. **Content**: This comparator calculates a weight on the edge based on the similarity content of the textual output of the two vertices connected via the edge.
3. **Directed content**: This comparator calculates the same similarity metric as the "Content" comparator, however it only analyses a subset of the communication, the communication being sent specifically between the vertices along the edge. (Thus the "Content" comparator would capture broadcasts e.g. tweets, and this comparator would not). However this weight is given greater significance.

3. Method

3.1. Algorithm

The general algorithm will be as follows, given as a input a list of activities where receiver can be n/a in the case of a broadcast. This can be as a stream or as a batch, as the algorithm will simply update it's model for each new activity.

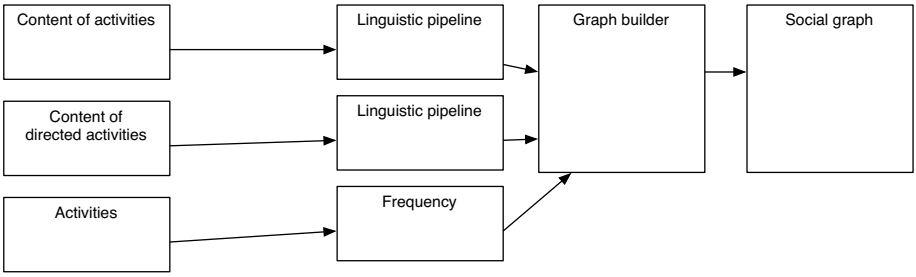

Figure 2. Functional System Architecture

The weight between two actors (vertices from now) is calculated in the following manner, the details of these two steps will be presented in subsections 3.3 and 3.2.

1. Calculate frequency of communication between any two vertices in the social graph.
2. Calculate the distance in content (keywords) between the textual output of every two vertices in the social graph. This step is actually carried out in two comparators (see section 2) of the functional system, on two different sets of input data, however the algorithm for comparison is equal.

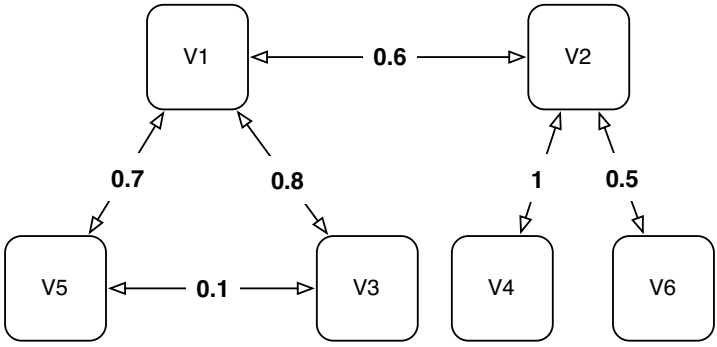

Figure 3. An example of a social network after weight calculation

After the network is pruned in the following manner to generate a sub-graph that captures subgroups:

1. Remove any edges with a weight below a certain threshold.
2. Apply the edge beetweenness algorithm removing N edges.

3.2. Content classification

This is done using GATE[19,20] textual analysis pipeline, ANNIE. ANNIE is applied much in the same way as in [21]. Using ANNIE we extract keywords from the a given. Thus we can concat all text from different vertices, and produce a set of keywords for

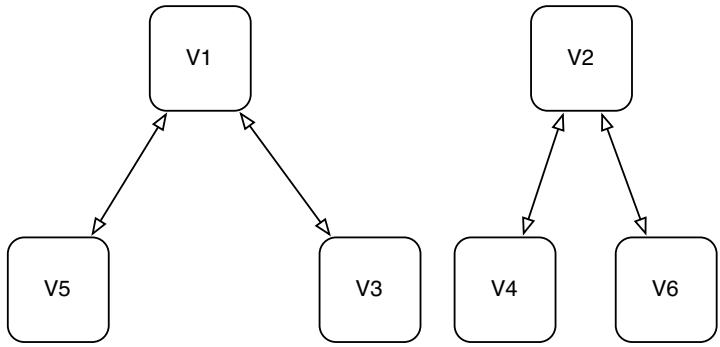

Figure 4. An example of a social network after pruning. In this pruning the first stage of the pruning (threshold) will remove the edge between vertices $V3$ and $V5$. The second stage of the pruning (betweenness) will remove the edge between vertices $V1$ and $V2$. the After this pruning two subgroups are identified; $[V1, V3, V5]$ and $[V2, V4, V6]$

Algorithm 1 Generate network, and prune according to threshold. After running this algorithm the social graph is given by the weight matrix in fw. The original weight matrix w is also kept for next update of the social graph, when new activities are added to the stream.

```
1:  for i = 0 to getSize(Graph) do
2:      for j = 0 to getSize(Graph) do
3:          if i! = j then
4:              for k = 0 to getSize(Comparators) do
5:                  w_ij = w_ij + runComparator(k, getVertex(i), getVertex(j)))
6:              end for
7:          end if
8:          if w_ij < threshold then
9:              fw_ij = 0
10:         else
11:             fw_ij = w_ij
12:         end if
13:     end for
14: end for
```

each vertex. This set of keywords can be compared against the set of the opposing vertex, producing a metric which is input for the weight of the edge between the vertices.

3.3. Graph analysis

After calculating the weighted graph and pruning the graph according to the threshold the edge-betweenness[3] algorithm, also called the Girvan-Newman algorithm[6], is applied. We chose this algorithm over its faster counterparts e.g. [7] and to an even greater extent [8] because of its implementation simplicity [4]

[3]The term betweenness in the context of graphs was introduced by [22]

[4]The algorithm is readily available through the JUNG library: `http://jung.sourceforge.net/`

4. Summary and Future Work

This is a work in progress and he current system is limited by the run time of the ANNIE system, which requires a substantial CPU time for text analysis. This sets a upper threshold for the number of social network vertices and activities per vertex the system can handle before becoming unresponsive. In future versions we plan on implementing a subset of the ANNIE features, thus reducing the CPU time required for keyword extraction.

In addition we plan on adding a semi-supervised learning algorithm for tuning of the important parameters such as weight threshold.

Finally the architecture of the system, closely resembling an ensemble system, enables developers to add additional comparators. Thus we look forward to experimenting with new measurements of "likeness" within social interaction.

5. Acknowledgments

The work presented here is supported by the European R&D project SOCIETIES[5].

6. Reproducibility

The work presented here is not bound by any dataset, the code that is based upon the concepts described in this paper is available as open-source (licensed under "FreeBSD" license) at this URL:
https://github.com/societies/SOCIETIES-Platform

References

[1] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998.
[2] J. Scott, *Social network analysis*. SAGE Publications Limited, 2012.
[3] J. Tyler, "Email as Spectroscopy: Automated Discovery of Community Structure within Organizations," *Arxiv preprint cond-mat/0303264*, 2003.
[4] M. Hönsch, "Detecting User Communities Based on Latent and Dynamic Interest on a News Portal," *Personalized Web-Science, Technologies and Engineering*, vol. 3, no. 2, pp. 47–50, 2011.
[5] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, no. 6825, pp. 268–276, 2001.
[6] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–6, Jun. 2002. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=122977&tool=pmcentrez&rendertype=abstract
[7] A. Clauset, M. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E*, vol. 70, no. 6, p. 066111, Dec. 2004. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevE.70.066111
[8] K. Wakita and T. Tsurumi, "Finding Community Structure in Mega-scale Social Networks," in *Proceedings of IADIS international conference on WWW/Internet.*, Feb. 2007, p. 9. [Online]. Available: http://arxiv.org/abs/cs/0702048
[9] A. Arenas, "Emergence of clustering, correlations, and communities in a social network model," *arXiv:cond-mat/0309263v2*, pp. 1–5, 2008.

[5]Project number: ICT-257493

[10] a. Arenas, L. Danon, a. Dï£¡az-Guilera, P. M. Gleiser, and R. Guimerï£¡, "Community analysis in social networks," *The European Physical Journal B - Condensed Matter*, vol. 38, no. 2, pp. 373–380, Mar. 2004. [Online]. Available: http://www.springerlink.com/openurl.asp?genre=article&id=doi:10.1140/epjb/e2004-00130-1

[11] C. Borgs and J. Chayes, "Exploring the Community Structure of Newsgroups [Extended Abstract]," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004.

[12] L. Backstrom, D. Huttenlocher, and J. Kleinberg, "Group Formation in Large Social Networks : Membership , Growth , and Evolution," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 44–54.

[13] J. R. Tyler, D. M. Wilkinson, and B. A. Huberman, "Email as spectroscopy: automated discovery of community structure within organizations," in *Communities and technologies*, M. Huysman, E. Wenger, and V. Wulf, Eds. Deventer, The Netherlands, The Netherlands: Kluwer, B.V., 2003, pp. 81–96. [Online]. Available: http://dl.acm.org/citation.cfm?id=966263.966268

[14] V. Gómez, A. Kaltenbrunner, and V. López, "Statistical analysis of the social network and discussion threads in slashdot," in *Proceedings of the 17th international conference on World Wide Web*, ser. WWW '08. New York, NY, USA: ACM, 2008, pp. 645–654. [Online]. Available: http://doi.acm.org/10.1145/1367497.1367585

[15] J. Bryden, S. Funk, and V. Jansen, "Word usage mirrors community structure in the online social network twitter," *EPJ Data Science*, vol. 2, no. 1, p. 3, 2013.

[16] D. Liu, D. Percival, and S. E. Fienberg, "User interest and interaction structure in online forums," in *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010*, W. W. Cohen and S. Gosling, Eds. The AAAI Press, 2010.

[17] T. Anwar and M. Abulaish, "Mining an enriched social graph to model cross-thread community interactions and interests," in *Proceedings of the 3rd international workshop on Modeling social media*. ACM, 2012, pp. 35–38.

[18] D. Project, "Activity Streams - a format for syndicating social activities around the web," 2013. [Online]. Available: activitystrea.ms

[19] H. Cunningham, "GATE, a general architecture for text engineering," *Computers and the Humanities*, vol. 36, no. 2, pp. 223–254, 2002.

[20] D. Maynard, V. Tablan, H. Cunningham, C. Ursu, H. Saggion, K. Bontcheva, and Y. Wilks, "Architectural elements of language engineering robustness," *Natural Language Engineering*, vol. 8, no. 2-3, pp. 257–274, 2002.

[21] T. Iofciu, "Finding Communities of Practice from User Profiles Based On Folksonomies," in *Proceedings of the 1st International Workshop on Building Technology Enhanced Learning solutions for Communities of Practice*, 2006, pp. 288–297.

[22] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, pp. 35–41, 1977.