Duplicate Removal for Overlapping Clusters: A Study Using Social Media Data

Amit Paul **Animesh Dutta**
Department of Computer Science and Engineering
National Institute of Technology Durgapur, West Bengal, India

March 26, 2019

Outline

- Background
- Motivation
- Related Work
- The Retweet and Reply Network
- Assumptions
- The Work
- The Proposed Algorithm
- The Modified Algorithm
- Result
- Discussion and Analysis
- Conclusion

Background

- Social Media such as Twitter possess huge amount of information.
- Network analysis using retweet links between users is both promising and challenging.
- Retweet link is formed when a user retweets a tweet of another user.
- The work here is based on the links between the users using both retweet and reply.
- The group or cluster community created by individual user, using retweets and reply links, are highly overlapping.

Motivation

- Retweet graph where source retweeted destination is neglected (Bild et al., 2015).
- Retweet and reply links are recent whereas follow and friendship network links are not as many followers remain inactive for days or months.
- Highly overlapping cluster communities make differentiation difficult which motivates the current work.
- Some application of using retweet network are recommending followers, recommending feeds for tweeting etc.

Motivation (Cont...)

- Is it possible to find suitable place of each individual user among all the cluster communities? X belongs to which cluster.

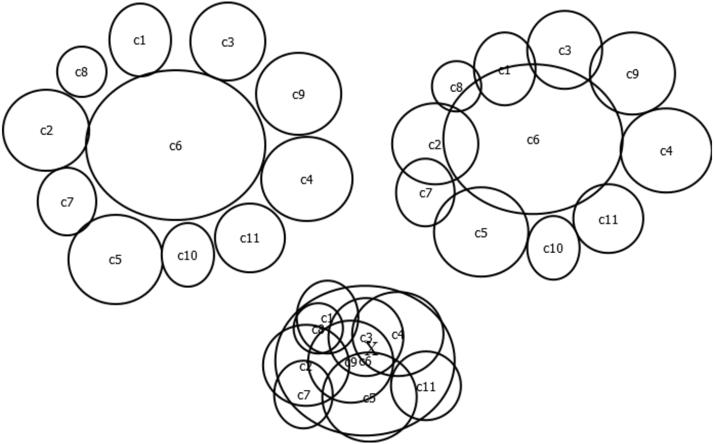

Figure: Overlapping cluster formation by individual user using retweet and reply links

Related Work

- Community clusters detection in social media is widely studied over the past decade (Zhang and Yu, 2015; Whang et al., 2016; Goldberg et al., 2010; Lee et al., 2010; Mishra et al., 2007).
- A user generally appears in more than one community and benchmark algorithms work better when overlapping is minimized (Lee et al., 2010).
- Community detection using modularity based methods is given by (Shiokawa et al., 2013; Clauset et al., 2004).
- (Lee et al., 2010; Whang et al., 2016) used seed expansion to detect overlapping communities.
- There is no clear understanding which technique is most suitable for a particular domain (Kloumann and Kleinberg, 2014) and the performance of community assignment algorithms (Lee et al., 2010).

Why People Retweet ?

- A retweet is a forwarded message from a user to his followers.
- A user in the Twitter network can retweet any other user's tweet.
- This shows the topical interest of the user who retweets the tweet of another user.

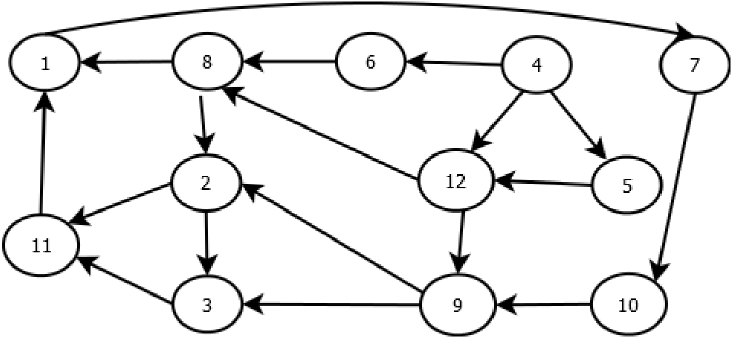

Figure: A retweet and reply network

The Retweet and Reply Network

- We conceptualize Twitter data in terms of a directed graph and is based on (Paul et al., 2016; Lussier and Chawla, 2011)
- The vertices represent users and the edges retweets or replies from one user to another.
- Each user called the target user, vertex in the graph, forms a group or community.
- The group is created in a breadth first manner, level-by-level, up to some pre-specified maximum level l (distance from start) .

The Retweet and Reply Network (Cont...)

- At each level l vertices or users are added to the cluster representing the target user.
- A set of clusters, a cluster configuration, is produced.
- Most of the groups created are overlapping, which necessitated duplicate user removal.
- The aim is to find the best suited place of a user (vertex) in a cluster among all the clusters in a set to create crisp clusters.
- Our approach, focuses on exact duplicate removal.

The Retweet and Reply Network(Cont...)

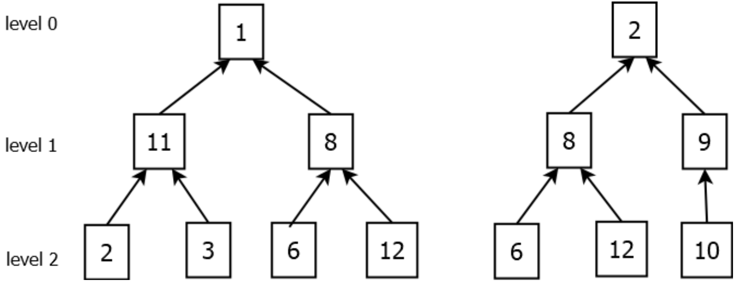

Figure: Cluster formed by target users 1 and 2

Assumptions

- The sideway links within same layer or level are not taken.
- A user appearing closest to the target user is kept only.
- There are no duplicate users within a cluster.

The Work

- Given a retweet graph $G = \{V, E\}$ where V is the vertex or user node and E is the directional edge.
- A user U_i is connected to another user U_j if U_j has retweeted or replied to user U_i, creating an edge E between U_j to U_i and is unidirectional.
- The experiments are performed on all the cluster formation and also by selecting the largest clusters defined using a threshold τ.
- The τ value is adjusted to give top 0.25%, 0.5%, 1.0%, 2.0%, 4.0% etc clusters of the total number of clusters for a predefined maximum level l.

The Proposed Algorithm

- The proposed algorithm deletes the duplicate users from the overlapping clusters.
- A empty bucket is taken which is populated by comparing a user in one cluster to the other.
- The user in the bucket is then used to delete the duplicates in the cluster set without any condition.

The Modified Algorithm

- A user U_i is more significant if it appears near to the root than the user further away from the root. U_e denotes a bucket member.
- The most significant user is placed in the bucket(E)after all comparisons.
  1. If $U_i = U_e$ and U_i (level) $< U_e$(level). Replace U_e by U_i.
  2. If $U_i \neq U_e$. Put U_i in E.
  3. If $E = \{\}$. Put U_i in E
- All the users those are least significant and $level \neq 0$ are deleted from the clusters.

Result

- The Geo-tagged Microblog data set available from the ARK data repository was used with 377616 tweets and 9477 users.
- 7123 clusters are formed for each level l, the remaining users have not received or send any retweet nor have replied or received any message.

Figure: Comparison of runtime of naive algorithm with proposed and modified algorithm for different levels$= 6, 7, 8, 9$. τ is set to 100%

Result

- Comparing duplicate using bucket set reduces runtime of algorithms compared to naive algorithm.

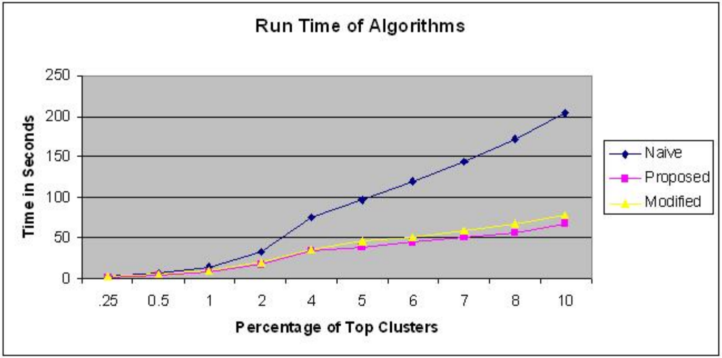

Figure: Comparison of runtime of naive algorithm with proposed and modified algorithm at level 9 with different τ values

Discussion and Analysis

- Top 10% of the clusters by size hold 33% of total users in the cluster set.
- Out of 7123 clusters only 15 clusters are of size more than 50 and only 4 clusters of size more than hundred at level 6.

Conclusion

- Here, the retweet and reply network among users are highly overlapping.
- This study shows the generation of crisp clusters using exact duplicate removal.
- One of the future work will focus on investigating the variation of physical distant between the users in the clusters.

THANK YOU

Bild, D. R., Liu, Y., Dick, R. P., Mao, Z. M., and Wallach, D. S. (2015). Aggregate characterization of user behavior in twitter and analysis of the retweet graph. *ACM Trans. Internet Technol.*, 15(1):4:1–4:24.

Clauset, A., Newman, M. E., and Moore, C. (2004). Finding community structure in very large networks. *Physical review E*, 70(6):066111.

Goldberg, M., Kelley, S., Magdon-Ismail, M., Mertsalov, K., and Wallace, A. (2010). Finding overlapping communities in social networks. In *2010 IEEE Second International Conference on Social Computing*, pages 104–113.

Kloumann, I. M. and Kleinberg, J. M. (2014). Community membership identification from small seed sets. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1366–1375, New York, NY, USA. ACM.

Lee, C., Reid, F., McDaid, A., and Hurley, N. (2010). Detecting highly overlapping community structure by greedy clique expansion. *ArXiv e-prints*.

Lussier, J. T. and Chawla, N. V. (2011). Network effects on tweeting. In *Proceedings of the 14th International Conference on Discovery Science*, DS'11, pages 209–220, Berlin, Heidelberg. Springer-Verlag.

Mishra, N., Schreiber, R., Stanton, I., and Tarjan, R. E. (2007). Clustering social networks. In *Proceedings of the 5th International Conference on Algorithms and Models for the Web-graph*, WAW'07, pages 56–67, Berlin, Heidelberg. Springer-Verlag.

Paul, A., Dutta, A., and Coenen, F. (2016). Cluster of tweet users based on optimal set. In *2016 IEEE Region 10 Conference (TENCON)*, pages 286–290.

Shiokawa, H., Fujiwara, Y., and Onizuka, M. (2013). Fast algorithm for modularity-based graph clustering. In *AAAI*, pages 1170–1176.

Whang, J. J., Gleich, D. F., and Dhillon, I. S. (2016). Overlapping community detection using neighborhood-inflated seed expansion. *IEEE Transactions on Knowledge and Data Engineering*, 28(5):1272–1284.

Zhang, J. and Yu, P. S. (2015). Community detection for emerging networks. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 127–135. SIAM.