

Uncovering Like-minded Political Communities on Twitter

Ophélie Fraisier

CEA, CEA-Tech Occitanie
IRIT, Université de Toulouse, CNRS
F-31062, Toulouse, France
ophelie.fraisier@cea.fr

Guillaume Cabanac

IRIT, Université de Toulouse, CNRS
F-31062, Toulouse, France
guillaume.cabanac@irit.fr

Yoann Pitarch

IRIT, Université de Toulouse, CNRS
F-31062, Toulouse, France
yoann.pitarch@irit.fr

Romarc Besançon

CEA, LIST
F-91191, Gif-sur-Yvette, France
romarc.besancon@cea.fr

Mohand Boughanem

IRIT, Université de Toulouse, CNRS
F-31062, Toulouse, France
mohand.boughanem@irit.fr

ABSTRACT

Stance detection systems often integrate social clues in their algorithms. While the influence of social groups on stance is known, there is no evaluation of how well state-of-the-art community detection algorithms perform in terms of detecting *like-minded communities*, i.e. communities that share the same stance on a given subject. We used Twitter’s social interactions to compare the results of community detection algorithms on datasets on the Scottish Independence Referendum and US Midterm Elections. Our results show that algorithms relying on information diffusion perform better for this task and confirm previous observations about retweets being better vectors of stance than mentions.

CCS CONCEPTS

• **Applied computing** → **Sociology**; • **Information systems** → *Web mining*;

KEYWORDS

Stance detection, Social media, Benchmarking

1 INTRODUCTION

The impact of communities and homophily¹ on the construction of people’s opinions have been studied by sociologists for decades. Bourdieu [3] noted that choosing between opinions usually meant choosing between groups supporting underlying opinions. Subsequent studies showed that these observations still held true on virtual social media, where users tended to interact with people sharing their opinions rather than debating with opponents, a phenomenon known as “*echo chambers*” [11, 13].

¹Homophily is the principle that a contact between similar people occurs at a higher rate than among dissimilar people.

This project is co-financed by the European Union – Europe is committed to Midi-Pyrénées with the European fund for regional development.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ICTIR’17, October 1–4, 2017, Amsterdam, The Netherlands.

© 2017 ACM. 978-1-4503-4490-6/17/10...\$15.00

DOI: <http://dx.doi.org/10.1145/3121050.3121091>

With the rapid growth of user-generated content, many researchers decided to built opinion mining systems, to automatically detect the subjective information users shared in their writings on the web [12]. These systems led to stance detection, which can be used in online social media to automatically determine the stance of specific users in favor or against a particular topic. Successful detection of stance can be very useful in various downstream tasks, such as information retrieval, text summarization or irony detection.

But an important question remains unanswered: can social interactions and communities alone be used to determine users’ stances? We decided to use the phenomenon of *echo chambers* and community detection to try to answer this question. This paper makes the following contributions: (a) a comparison of state-of-the-art community detection algorithms, testing their ability to uncover underlying groups who share a common stance and (b) a comparison between the potential of retweet, mention, and their combination graphs, as a means to uncover like-minded communities.

2 RELATED WORK

The relationship between opinions and social relations has been exploited in opinion mining for many years, with researchers trying to deepen their analysis by using social theories [20]. Some researchers, while not formally using community detection, integrated some aspects of it to improve their opinion classification systems. Speriosu et al. [19] used for example label propagation along the follower graph to assign polarity labels to tweets. On the other hand, opinion was sometimes used as a feature to detect like-minded community in conjunction with classical algorithms. Dinsoreanu and Potolea [9] exploited opinions extracted from documents and the infomap community detection algorithm to detect consistent communities of users.

Several benchmarks compared performances on ground-truth communities and some systems are explicitly made for detecting like-minded communities – like Deepak et al. [8] who used a bottom-up hierarchical clustering approach – but we did not find a benchmark focusing on how classical algorithms performed when detecting like-minded communities.

3 EXPERIMENTAL SETUP

3.1 Definitions

In this paper, we define a *community* as a group of users interacting for enough time and with enough commitment for connections to appear in the cyberspace [17].

The *stance* of a user is its publicly stated viewpoint on a particular subject. In our experiments, we focused on datasets with political viewpoints. Each dataset contains N stances, with $N \geq 2$. A stance may be shared by users belonging to different communities, and a community may contain several stances. Our aim is to detect if some algorithms obtain better performances in finding communities as homogeneous as possible in terms of stances, that is communities where a large majority of users share the same stance. We will call them *like-minded communities*.

3.2 Algorithms

We chose to compare in this study some popular and easily-accessible graph-based community detection algorithms², implemented in the *igraph* library [6]. Table 1 lists some characteristics of the algorithms presented below.

Table 1: Characteristics of community detection algorithms, including complexity on typical sparse data – W stands for edges weights and D for edges direction.

Family	Algorithm	W	D	Complexity
<i>Modularity maximisation</i>	FAST-GREEDY [22]	✓	✗	$O(mn \log n)$
	LEADING EIGENVECTORS [14]	✗	✗	$O(n^2)$
	MULTI-LEVEL [1]	✓	✗	$O(n)$
<i>Information diffusion</i>	INFOMAP [18]	✓	✓	$O(n(n+m))$
	LABEL PROPAGATION [16]	✓	✗	$O(n+m)$
<i>Random walk</i>	WALKTRAP [15]	✓	✗	$O(n^2 \log n)$

Modularity maximisation. The modularity measures the number of intra-community links versus links joining communities. FAST-GREEDY attempts to maximize modularity of the community structure by merging pairs of communities with a bottom-up approach. LEADING EIGENVECTORS uses a matrix-based approach analogous to a spectral partitioning method. MULTI-LEVEL iteratively merges partitions representing local maxima of modularity at different scales.

Information diffusion. INFOMAP attempts to create communities according to the flow of information present in the network. LABEL PROPAGATION is a simple iterative process where, at each step, each node adopts the label most present in its neighbors.

Random walk. WALKTRAP detects communities based on a distance that quantify the structural similarity between vertices using random walks.

3.3 Interactions

In this work we consider only the two main tweet-level interactions³: retweets (RT) and mentions (@). Retweeting means sharing another user’s tweet with one’s followers while mentioning is citing another user’s username. Three graphs are built per dataset: one per interaction plus one taking into account both interactions indifferently. In these graphs, vertices are users and edges are interactions. The weight of the edges are the number of times the

²We used non-parametric algorithms with respect to the number of communities to detect since, as expressed in Section 3.1, the ideal situation for us is not having one unique community per stance, but simply having homogeneous communities.

³We did not consider the *Follow* interaction because it is a user-level interaction, so hardly comparable with retweets and mentions, and it shows a more passive way of participating on Twitter.

users interacted. A vertex can represent an annotated or a non-annotated user: annotated users are those for which we know the stance, whereas non-annotated users appear in annotated users’ tweets because of mentions or retweets, but we do not know their stance. When a directed graph is required by the algorithm, we follow the information path: from retweeted users to users sharing tweets, and from mentioning users to users being cited.

3.4 Scoring functions

To compare the performance of algorithms on like-minded communities we used the *purity* [10] and the *normalized mutual information* [7], defined as follows:

$$\text{Purity}(\Omega, \mathbb{C}) = \sum_k \frac{\max_j |\omega_k \cap c_j|}{N} \quad \text{NMI}(\Omega, \mathbb{C}) = \frac{2 \times I(\Omega, \mathbb{C})}{H(\Omega) + H(\mathbb{C})}$$

with Ω the partition of detected communities, \mathbb{C} the set of stances, ω_k the set of annotated users in community k , c_j the set of annotated users in class j , N the total number of annotated users in P , I the mutual information, and H the entropy.

Purity is known to be biased in favor of small communities but in our case, since we want communities as homogeneous as possible, it is still a valid indicator. To examine the consistency of the community detection, we also used the *standard deviation* (SD) of the intra-communities purity scores.

4 EXPERIMENTS

4.1 Datasets

The use of community detection algorithms as a mean to detect like-minded communities forces us to have a non-negligible amount of interactions between the users. For this reason we cannot use datasets built around keywords, like the ones featured in the task 6 of SemEval2016⁴ because users in this kind of datasets seldom interact with one another. We used two datasets published by Brigadir et al. [4], both constructed by gathering all available tweets for a fixed subset of users (see Table 2). DMI-TCAT [2] collected the tweets from Twitter API, based on the released users accounts and tweets ids.

Scottish Independence Referendum (SR). This dataset was originally collected from Aug 11th to Oct 19th 2014. It reflects the debate around the Scottish Independence referendum of the 18th of September 2014. The official “Yes” and “No” campaigns were very active on social media. To be included as a “Yes” or “No” supporter, users had to be part of the Scottish Independence Referendum Electoral Commission or unambiguously self-identify as such on their profile.

Table 2: Dataset sizes⁵

Dataset	Stance	Original		Retrieved	
		Users	Tweets	Users	Tweets
SR	Yes	618	799,096	535	344,563
	No	610	570,024	508	263,569
ME	Democrat	942	89,296	701	56,671
	Republican	997	80,840	756	56,506

⁴<http://alt.qcri.org/semeval2016/task6/>

⁵The differences between the original sizes and the retrieved ones are due to users having deleted their tweets and their accounts since the initial crawl.

US Midterm Elections (ME). This dataset concerns the US Midterm elections held on the 4th of November 2014 and was originally collected from Oct 10th to Nov 20th 2014. Third parties were ignored so users are either “Democrat” or “Republican”. They were selected thanks to several sources listing official Twitter accounts of campaigners in these elections.

4.2 Scores on all users

Table 3 presents some characteristics of the resulting graphs. Given the construction method for the graphs, we do not have annotated users in all communities. In order to measure consistent scores, we exclusively consider communities having at least 3 annotated users (see Table 4). We will discard from the observations below the scarce cases where we only have one community to analyse, since standard deviation and NMI cannot be calculated on a unique community.

Table 5 presents the scores by interaction and algorithm. The first observation we can make is that retweets appear to be more important in terms of stance diffusion. For retweets, while almost all algorithms obtain a purity score of more than 90%, LABEL PROPAGATION and INFOMAP obtain smaller standard deviation values. LABEL PROPAGATION also obtain the best NMI scores for both dataset. For mentions, results are more variable. INFOMAP, FAST-GREEDY and

Table 3: Characteristics of the constructed graphs – D is the density measure and C the clustering coefficient.

		# Vertices	# Edges	Degrees			D	C
				Mean	Med	Max		
<i>Using interactions between all users</i>								
SR	RT	78,854	266,146	7	1	7,391	10^{-5}	0.019
	@	59,122	200,631	7	1	1,721	10^{-4}	0.044
	Both	120,165	443,322	7	1	7,785	10^{-5}	0.031
ME	RT	149,137	291,137	4	1	10,125	10^{-5}	0.001
	@	23,148	35,141	3	1	2,278	10^{-4}	0.004
	Both	163,646	320,007	4	1	10,288	10^{-5}	0.001
<i>Using interactions between annotated users only</i>								
SR	RT	898	16,938	38	21	302	0.042	0.298
	@	258	278	2	1	237	0.008	0.002
	Both	902	16,938	38	21	302	0.041	0.298
ME	RT	973	2,056	4	2	69	0.004	0.134
	@	126	125	2	1	122	0.015	0.000
	Both	989	2,180	4	2	123	0.004	0.106

Table 4: Number of communities containing at least 3 annotated users – the algorithms are represented by their initials.

Algorithm	Scottish Referendum			Midterms Elections		
	RT	@	Both	RT	@	Both
<i>Using interactions between all users</i>						
Modularity maximisation	F	15	34	27	39	45
	LE	19	1	1	3	19
	M	17	36	29	43	47
Information diffusion	I	102	85	38	112	159
	LP	4	2	3	74	135
Random walk	W	15	34	27	39	45
<i>Using interactions between annotated users only</i>						
Modularity maximisation	F	4	2	4	49	2
	LE	4	2	2	33	2
	M	4	2	5	46	2
Information diffusion	I	138	2	137	111	2
	LP	4	2	4	94	2
Random walk	W	4	2	4	49	2

WALKTRAP obtain good performances. Taking into account both interactions does not seem to offer an advantage, since purity and NMI scores for these graphs are inferior to those of the retweet graphs. All in all, algorithms based on information diffusion seem to have a slight advantage. They consistently detect communities with a high purity, and despite the fact that they detect more communities, they still obtain better NMI scores than the other algorithms most of the time. For all graphs, MULTI-LEVEL and LABEL PROPAGATION are the quickest algorithms with a runtime under 2 seconds. The slowest ones are FASTGREEDY and WALKTRAP, with a runtime up to 25 minutes for the Scottish Independence Referendum dataset, and up to 75 minutes for the US Midterm Elections dataset. These observations are not really surprising given the respective complexities of the algorithms (see Table 1).

4.3 Scores on annotated users only

One important caveat of this analysis is of course the communities without annotated users. Indeed, since all interactions are used to

Table 5: Scores for the SR and ME datasets – best scores are presented in bold and cases for which we have one community only to analyse in gray and in italic (see table 4).

Algorithm	Scottish Referendum			Midterms Elections			
	RT	@	Both	RT	@	Both	
<i>Using interactions between all users</i>							
Purity	Modularity maximisation	F	0.97	0.78	0.86	0.93	0.63
		LE	0.87	<i>0.51</i>	<i>0.51</i>	0.62	0.58
		M	0.97	0.72	0.84	0.87	0.66
Information diffusion	I	0.96	0.80	0.88	0.97	0.79	
	LP	0.98	0.53	0.54	0.93	0.78	
Random walk	W	0.97	0.78	0.86	0.93	0.63	
SD	Modularity maximisation	F	0.16	0.13	0.15	0.14	0.12
		LE	0.15	<i>n/a</i>	<i>n/a</i>	0.24	0.15
		M	0.17	0.15	0.16	0.17	0.12
Information diffusion	I	0.10	0.19	0.17	0.10	0.18	
	LP	0.02	0.33	0.24	0.09	0.17	
Random walk	W	0.16	0.13	0.15	0.14	0.12	
NMI	Modularity maximisation	F	0.55	0.12	0.25	0.28	0.02
		LE	0.33	<i>n/a</i>	<i>n/a</i>	0.14	0.023
		M	0.52	0.08	0.21	0.22	0.04
Information diffusion	I	0.22	0.13	0.21	0.24	0.11	
	LP	0.85	0.01	0.01	0.25	0.11	
Random walk	W	0.55	0.12	0.25	0.28	0.02	
<i>Using interactions between annotated users only</i>							
Purity	Modularity maximisation	F	0.99	0.76	0.99	0.98	0.58
		LE	0.98	0.76	0.86	0.86	0.58
		M	0.99	0.76	0.98	0.98	0.58
Information diffusion	I	0.96	0.75	0.97	0.97	0.58	
	LP	0.99	0.74	0.98	0.98	0.58	
Random walk	W	0.99	0.76	0.99	0.98	0.58	
SD	Modularity maximisation	F	0.01	0.06	0.01	0.06	0.10
		LE	0.02	0.06	0.16	0.07	0.06
		M	0.01	0.06	0.01	0.06	0.06
Information diffusion	I	0.09	0.09	0.09	0.08	0.06	
	LP	0.01	0.09	0.01	0.08	0.06	
Random walk	W	0.01	0.06	0.01	0.06	0.06	
NMI	Modularity maximisation	F	0.88	0.18	0.86	0.32	0.001
		LE	0.69	0.18	0.50	0.26	0.001
		M	0.84	0.18	0.81	0.32	0.001
Information diffusion	I	0.22	0.15	0.23	0.24	0.001	
	LP	0.86	0.15	0.85	0.27	0.001	
Random walk	W	0.88	0.18	0.86	0.32	0.001	

construct the retweets and mentions graphs, we have a majority of users for whom we do not know the stance, and therefore a lot of communities for which we cannot evaluate stance homogeneity since they do not contain enough annotated users. To overcome this issue, we opted for detecting communities on the sub-graphs representing the interactions between annotated users only (see Table 3 for sub-graphs characteristics).

The results (see Table 5) show again that, even when using only the interactions between the subset of users used to build the datasets, stance can be determined by community.⁶ By focusing on annotated users only, we obtain for almost all graphs better scores. Interestingly, for retweets, algorithms based on information diffusion do not seem to have an advantage over the others, contrary to the previous section. This may be due to the fact that focusing on interactions between annotated users does not allow to properly model the diffusion of information.

5 DISCUSSION

Retweet graphs generally seem to be a better way to detect homogeneous like-minded communities than mention graphs. This is consistent with the previous observations made in the literature so far [5].

When considering retweet interactions, the leading algorithms are LABEL PROPAGATION and INFOMAP, two algorithms relying on information diffusion. This is a valuable observation, suggesting that, in a certain way, stance “follows” information on Twitter. These observations suggest that to efficiently detect like-minded communities on Twitter, it is a better idea to look at the information circulating between users rather than considering purely structural criteria. However the direction of diffusion does not seem to be important, since the implementation of label propagation we used in this work considers undirected edges. For mentions however, we can see that these algorithms do not really have an advantage compared to others, suggesting that mention graphs do not have the same underlying principles that retweet graphs, stressing a difference in use between these interactions.

This work featured only two interactions taken separately and focuses mainly on atypical users, limiting the generalization capability of this study. The formation and diffusion of stances is a much more complex process, still hard to grasp by computing methods. We may assume that every interactions on social media has its own role in this process. It would be interesting to expand this comparison to more algorithms, datasets (including some with more than 2 stances), and interactions, and to see how well the systems taking into account several interactions [21] perform on like-minded communities.

6 CONCLUSION

In this article, we compare the performances of three classes of community detection algorithms for the detection of like-minded communities on Twitter. Despite the growing integration of social features in opinion mining systems, there is no existing comparison of state-of-the-art community detection methods on this particular task.

⁶We must note that LEADING EIGENVECTOR does not succeed in extracting communities on the graph taking into account both interactions (see Table 4).

We found out that algorithms based on information diffusion seem to perform better when using retweet interactions and upheld the fact that retweets are a better vector of opinion than mentions. Overall, label propagation seems to be a good choice for detecting like-minded communities, it achieves excellent results and is one of the fastest among the algorithms we benchmarked.

Our results confirms that popular community detection algorithm can indeed be used as-is to reliably detect users’ stance under certain conditions.

REFERENCES

- [1] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast Unfolding of Communities in Large Networks. *JSTAT* 10 (2008), P10008. DOI: <http://dx.doi.org/10.1088/1742-5468/2008/10/P10008>
- [2] Erik Borra and Bernhard Rieder. 2014. Programmed Method: Developing a Toolset for Capturing and Analyzing Tweets. *AJIM* 66, 3 (2014), 262–278. DOI: <http://dx.doi.org/10.1108/AJIM-09-2013-0094>
- [3] Pierre Bourdieu. 1973. L’opinion Publique n’existe pas. *Les temps modernes* 318 (1973), 1292–1309.
- [4] Igor Brigadir, Derek Greene, and Pádraig Cunningham. 2015. Analyzing Discourse Communities with Distributional Semantic Models. In *WebSci*. 1–10. DOI: <http://dx.doi.org/10.1145/2786451.2786470>
- [5] M. D. Conover, J. Ratkiewicz, M. Francisco, B. Gonçalves, A. Flammini, and F. Menczer. 2011. Political Polarization on Twitter. In *ICWSM*. 89–96.
- [6] Gabor Csardi and Tamas Nepusz. 2006. The Igraph Software Package for Complex Network Research. *InterJournal Complex Systems* (2006), 1695.
- [7] Leon Danon, Albert Diaz-Guilera, Jordi Duch, and Alex Arenas. 2005. Comparing Community Structure Identification. *JSTAT* 09 (2005), P09008. DOI: <http://dx.doi.org/10.1088/1742-5468/2005/09/P09008>
- [8] Talasila Sai Deepak, Hindol Adhya, Shyamal Kejrival, Bhanuteja Gullapalli, and Saswata Shannigrahi. 2016. A New Hierarchical Clustering Algorithm to Identify Non-Overlapping Like-Minded Communities. In *HT*. 319–321. DOI: <http://dx.doi.org/10.1145/2914586.2914613>
- [9] Mihaela Dinsoreanu and Rodica Potolea. 2015. Opinion-Driven Communities’ Detection. *IJWS* 10, 4 (2015), 324–342. DOI: <http://dx.doi.org/10.1108/IJWS-04-2014-0016>
- [10] M. Girvan and M. E. J. Newman. 2002. Community Structure in Social and Biological Networks. *NAS* 99, 12 (2002), 7821–7826. DOI: <http://dx.doi.org/10.1073/pnas.122653799>
- [11] Shanto Iyengar and Sean J. Westwood. 2015. Fear and Loathing across Party Lines: New Evidence on Group Polarization. *AJPS* 59, 3 (2015), 690–707. DOI: <http://dx.doi.org/10.1111/ajps.12152>
- [12] Bing Liu. 2012. Sentiment Analysis and Opinion Mining. *Synthesis Lectures on HLT* 5, 1 (2012), 1–167. DOI: <http://dx.doi.org/10.2200/S00416ED1V01Y201204HLT016>
- [13] Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology* 27, 1 (2001), 415–444. DOI: <http://dx.doi.org/10.1146/annurev.soc.27.1.415>
- [14] M. E. J. Newman. 2006. Finding Community Structure in Networks Using the Eigenvectors of Matrices. *PRE* 74, 3 (2006). DOI: <http://dx.doi.org/10.1103/PhysRevE.74.036104>
- [15] Pascal Pons and Matthieu Latapy. 2005. Computing Communities in Large Networks Using Random Walks. In *ISCI*. Vol. 3733. 284–293. DOI: http://dx.doi.org/10.1007/11569596_31
- [16] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. 2007. Near Linear Time Algorithm to Detect Community Structures in Large-Scale Networks. *PRE* 76, 3 (2007). DOI: <http://dx.doi.org/10.1103/PhysRevE.76.036106>
- [17] Howard Rheingold. 1993. *The virtual community: Finding connection in a computerized world*. <http://www.rheingold.com/vc/book/>
- [18] M. Rosvall, D. Axelsson, and C. T. Bergstrom. 2009. The Map Equation. *EPJ ST* 178, 1 (2009), 13–23. DOI: <http://dx.doi.org/10.1140/epjst/e2010-01179-1>
- [19] Michael Speriosu, Nikita Sudan, Sid Upadhyay, and Jason Baldridge. 2011. Twitter Polarity Classification with Label Propagation over Lexical Links and the Follower Graph. In *EMNLP*. 53–63.
- [20] Jiliang Tang, Yi Chang, and Huan Liu. 2014. Mining Social Media with Social Theories: A Survey. *ACM SIGKDD Explorations Newsletter* 15, 2 (2014), 20–29. DOI: <http://dx.doi.org/10.1145/2641190.2641195>
- [21] Lei Tang, Xufei Wang, and Huan Liu. 2012. Community Detection via Heterogeneous Interaction Analysis. *DMKD* 25, 1 (2012), 1–33. DOI: <http://dx.doi.org/10.1007/s10618-011-0231-0>
- [22] Ken Wakita and Toshiyuki Tsurumi. 2007. Finding Community Structure in Mega-Scale Social Networks. In *WWW*. 1275. DOI: <http://dx.doi.org/10.1145/1242572.1242805>