Investigating Similarity of Nodes' Attributes in Topological Based Communities.

Rajesh Sharma
University of Tartu
Tartu, Estonia
rajesh.sharma@ut.ee

Danilo Montesi
University of Bologna
Bologna, Italy
danilo.montesi@unibo.it

ABSTRACT

One of the important problems in the domain of network science is the community detection. In the past, various topological based community detection algorithms have been proposed. Recently, researchers have taken into account attributes of the nodes while proposing community detection algorithms. In this work, we investigate *if the nodes in a community, identified through topology based algorithms also exhibit attribute similarity.* Using four different kinds of similarity metrics, we analyse the attribute similarity of the nodes within the communities derived using five different types of topological based community detection algorithms. Based on our analysis of three real social network datasets, we found on an average of 50% attribute similarity among the nodes in the communities.

KEYWORDS

Social network analysis; Community analysis; Attributed networks.

ACM Reference Format:
Rajesh Sharma and Danilo Montesi. 2018. Investigating Similarity of Nodes' Attributes in Topological Based Communities.. In *WWW '18 Companion: The 2018 Web Conference Companion, April 23–27, 2018, Lyon, France*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3184558.3191564

1 INTRODUCTION

One of the most investigated work in the domain of networks is the analysis of community structures [8]. A lot of interesting and seminal approaches have been proposed, which are based on the topology of the networks. For example, in [4] authors presented a hierarchical agglomeration algorithm, based on link structure. In addition, various improvements have been proposed especially for scalability [23]. The basic idea behind all these approaches is that in a network, the nodes which are more densely connected with each other are part of a community compared to rest of the nodes. For more intensive surveys regarding the topological based community detection approaches, readers can refer to [8], [6], and [24].

The topology based algorithms only consider the structure of the network and ignores the attributes of the nodes while identifying communities. Examples of attributes include geographic location, interests, affiliations, etc. These attributes[1] can be a reason to motivate users to align with a particular set of nodes in a network, and form communities. For example, consider the famous Zachary club dataset, which is often the test bed for various community detection algorithms (Figure 2(a), [8]). The two main communities were formed because of the opinion of the club members regarding the fee structure with respect to the views of the president and the instructor. Thus, if opinion can be considered as an example of attribute, then based on the above example, one can argue that attribute (in this case the opinion) plays an important role in community formation. However, that may not be the case for each community formation.

Recently, there has been interest among the researchers in the domain of attribute based communities. In this line of research, works include proposal of i) detecting communities by using the attribute information of the nodes [9] as well as analysing community structures solely based on attributes [22], or ii) hybrid approach, which exploits both topology of networks and the attributes of the nodes [17], [25], [18], [14], [5] and [1].

In a different set of works, which are closer to that of ours, researchers have evaluated various community detection algorithms by using a priori information about the communities in the networks [12], [13], [19], [25]. The objective of this work is different, as we investigate nodes' similarity in the communities, which are identified through topology based community detection algorithms.

Let $G{<}N, E, I{>}$ represents a network, where N represents the total nodes, E the total edges, and I as the total set of interests of all the nodes in the network G. Assuming C represents the set of communities formed from a network G, by using a topological based community detection algorithm. Let $C_i{<}N_i', E_i', I_i'{>} \in C$, (where $N^{'} \subseteq N$, $E^{'} \subseteq E$ and $I^{'} \subseteq I$), represents a community identified through some topological based community detection algorithms.

To understand the similarity among the nodes, we have used four different types of similarity metrics which cover different perspectives of similarities (see Section III). Let S represents the similarity metric set. We calculate the similarity score among the neighboring nodes $\in N^{'}$ using each S_j

WWW '18 Companion, April 23–27, 2018, Lyon, France

ACM ISBN 978-1-4503-5640-4/18/04.
https://doi.org/10.1145/3184558.3191564

[1] From here on we will use the term interest and attribute interchangeably.

$\in S$, to understand how much attribute similarity is present among the nodes of a particular community.

To the best of our knowledge, no work in the past has investigated attribute similarity of the nodes within the topological based communities. We study this problem by exploring three different real social network datasets, namely i) Blippr (product rating social network), ii) Last.fm (music based social network) and iii) Delicious (social bookmarking website). These three datasets have social network as well as have attribute information of the nodes in them. In contrast to the past studies [11], [15] based on the common belief that *birds of the same feather flocks together*, we found out that there exists a very low attribute similarity among the nodes of a community.

The rest of the paper is organized as follows. Next, we describe the datasets. Section III presents our results related to community analysis. We conclude with a discussion of future directions in Section IV.

2 DATASETS

In this section, we briefly describe the datasets we analysed.
Blippr: It is a social network[2], where users can provide comments and reviews for various products and categories. Users can also link with other users as friends. For this dataset, we constitute five main categories, namely i) Movies/TV, ii) Technology (softwares, gadgets, etc.), iii) Music Album, iv) Video Games, and v) Books.
Last.fm: This dataset has been collected from Last.fm[3], which is a social network platform which allows tagging facility for the users, with respect to the artists. Users can also become friends with other users. In this dataset, the artists to which users listen is considered as attributes.
Delicious: Delicious[4] is a social network platform, where users can tag their favorite links along with creating social links with other users. In this dataset, the attribute is considered as the web links. Delicious and the Last.fm dataset was released by researchers at [3].

These particular datasets have been selected as the social network and attribute information of the nodes have not necessarily influenced each other unlike in DBLP co-authorship, where edge formation gets created due to a common interest (for example, publications in the similar domain).

Table 1 enlists various network properties of the three datasets. All the three datasets have ∼2000 nodes. However, the total edges in these three datasets vary a lot. In spite of Last.fm having more edges than Delicious, the clustering coefficient of Delicious is higher. Also, the diameter (and average path length) of Delicious is much larger than that of Last.fm and Blippr. In the last row, we also provide information about the total number of interests for each network. All the three datasets show heavy-tailed degree distribution (Figures not shown due to space limitation).

Properties/Datasets	Blippr	Delicious	Last.fm
Number of Nodes	1944	1861	1892
Total Edges	9040	15328	25434
Clustering Coefficient	0.015	0.41	0.13
Diameter	7	16	9
Average Path Length	2.29	5.36	3.51
Number of Interests	5	104799	17632
Average interests per node	2	56	49

Table 1: Datasets: Basic properties.

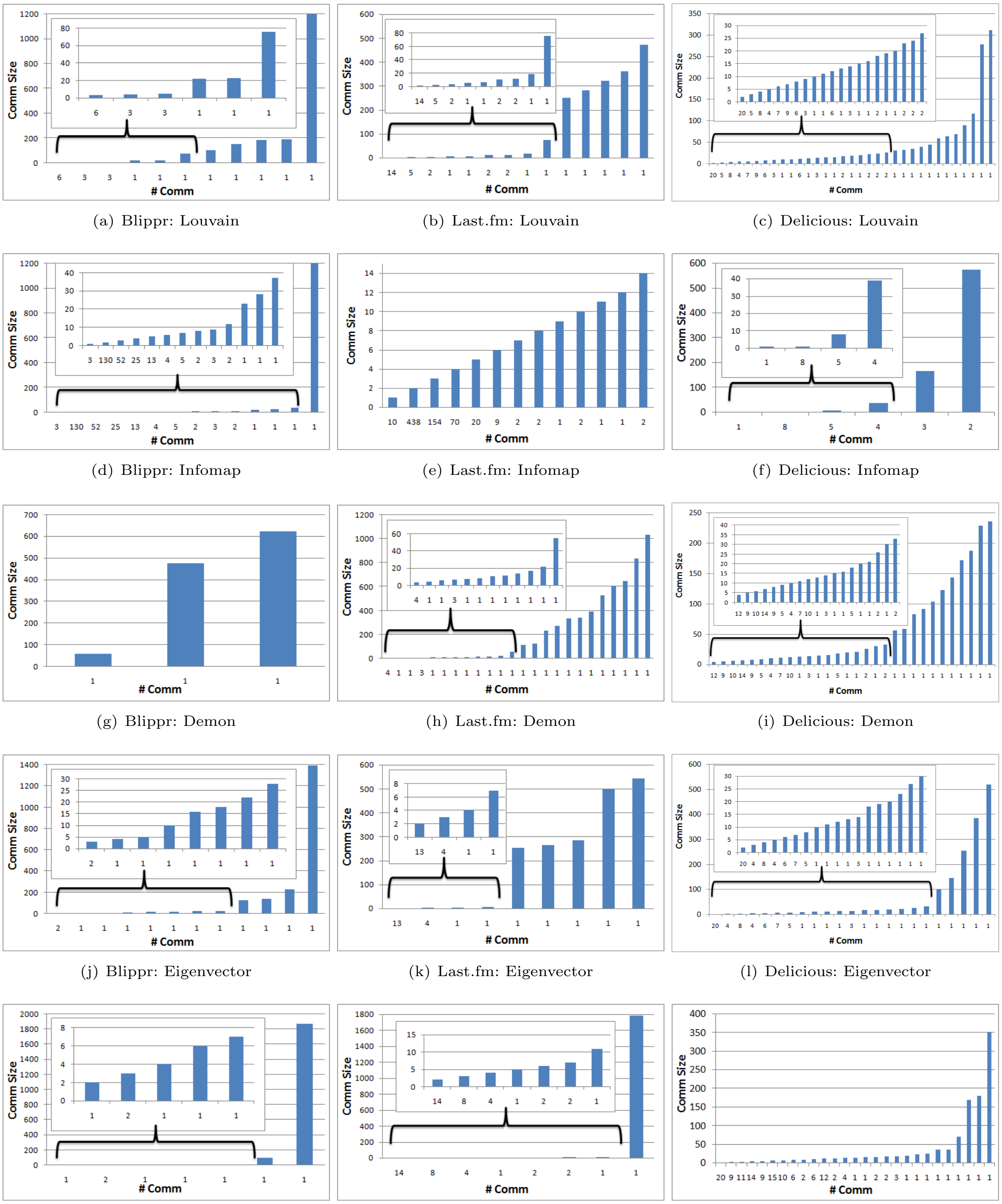
3 EXPERIMENTS

We first discuss the five community detection algorithms which are used for identifying communities. Later we describe four metrics we used for calculating the similarity among the nodes in a community. We conclude this section, with the discussion about the results.

3.1 Community Detection Algorithms

The five community detection algorithms to generate communities are following:
1. Louvain: Based on the greedy modularity approach, the algorithm tries to optimize the modularity in a two step process. In the first step, small communities are identified and in the second step it collects nodes belonging to the same community and create new networks from the nodes denoting the communities. It is one of the fastest community detection algorithms and can handle very large networks. However, it often fails to produce communities of medium size [2].
2. Infomap: The method is flow-based which exploits the maps of random walks [21]. The approach uses the probability flow of random walks as a medium to extract communities by squeezing the path of walks to represent communities. The algorithm is also able to handle weighted and directed networks.
3. Demon: This algorithm uses a recursive approach to collect denser areas from ego-networks. Thus, the communities produce by this algorithm have higher internal density. As the method uses ego-networks as the starting point for community identification, the resultant communities can be overlapping in nature [7].
4. Leading Eigenvector: The crux of the algorithm is the modularity matrix of the form A-P, where A is the adjacency matrix of the network and P represents the probability matrix for the edges between all the nodes of the network, considering it as a random network. The algorithm starts by calculating the eigenvector of the modularity matrix for the largest positive eigenvalue and then separating vertices into two communities based on the sign of the corresponding element in the eigenvector [16].
5. Label Propagation: Initially, this algorithm assigns every node with a different label. At each iterative step, a node picks a most popular label among the labels of neighbors. Eventually, the nodes agrees on a label and form a part of a community. This iterative and simple strategy helps the

[2] It has now been acquired by Mashable (http://mashable.com/).
[3] http://www.last.fm
[4] http://www.delicious.com



(a) Blippr: Louvain

(b) Last.fm: Louvain

(c) Delicious: Louvain

(d) Blippr: Infomap

(e) Last.fm: Infomap

(f) Delicious: Infomap

(g) Blippr: Demon

(h) Last.fm: Demon

(i) Delicious: Demon

(j) Blippr: Eigenvector

(k) Last.fm: Eigenvector

(l) Delicious: Eigenvector

(m) Blippr: Label Prop

(n) Last.fm: Label Prop

(o) Delicious: Label Prop

Figure 1: Community distribution of the datasets.

Comm. Det Algo	Blippr	Delicious	lastfm
Louvain	20	34	34
Infomap	243	713	712
Demon	3	24	29
Leading Eigenvector	13	24	24
Label Propagation	8	34	33

Table 2: Community Size information

algorithm in achieving a complexity which is linear in terms of execution time [20].

The above five community detection algorithms are diverse in nature as they follow different strategies to detect communities. For example, Demon produces overlapping communities, whereas other three generate disjoint communities. Louvain, Leading eigenvector, Infomap algorithms are much faster compared to Label propagation and Demon. Algorithms such as Label propagation are based on a bottom up approach, whereas Louvain uses a top-down methodology to detect communities.

Table 2 provides information about the number of communities generated through each of the community detection algorithms for each of the datasets. For Delicious and Last.fm, all the community detection algorithms have generated almost same number of communities. In Figure 1, for each dataset and for each of the community detection algorithms, we provide detailed information about the number of communities with a particular size. In all the cases, the resultant number of communities shows heavy tail distribution. For our analysis, we have only considered the communities which are bigger than of size two.

3.2 Measurements

We now describe the similarity metrics which we used to analyse the similarity among the nodes of a community. These following metrics cover different aspects of similarity and cannot be considered as a replacement for one another.
1. Sim: We call our basic similarity measurement as *Sim*. For any two nodes n_x and n_y, Sim(I_{n_x},I_{n_y}) returns 1 if at least one common interest is present among any two neighboring nodes. That is if $|I_x \cap I_y| \geq 1$, Sim(I_{n_x},I_{n_y}) returns 1 otherwise 0.
2. Jaccard: *Sim* has binary output. To cover the relative aspect of similarity, we measure the Jaccard similarity between every two connected nodes within communities. Formally, it is defined as ratio of $|I_x \cap I_y|$ and $|I_x \cup I_y|$.
3. Cosine: Jaccard is a cardinality based measurement. To cover the degrees of similarity between two connected nodes, we measure the cosine similarity between the nodes, which is defined as the ratio of the dot product of the interest vectors (I_x. I_y) and magnitude ($||I_x||\ ||I_y||$).
4. EI Index: Compared to the above three measurements which are edge based, we also measure a similarity metric based on the ego network of the nodes, which is based on the External-Internal (EI) Index [10]. In the original definition, within a community, an edge with respect to a node is called internal if the other node connected with the edge is also within the same community otherwise it is called external. The modified definition in our analysis is as follows. For any community, the meaning of external and internal has to do with the interest and we only consider the edges of an ego network of the nodes which are present in the community under investigation. In other words, within a community, an edge for a node is considered internal if the other node has at least one common interest with respect to the ego node otherwise external. Unlike in the original EI index, the EI index is measured as the ratio of the difference between the total internal edges and total external edges to the union of external and internal edges. This is done for normalization purpose. In the above three metrics, the value of the similarity function lies between 0 and 1, where higher is the value the more similar two nodes are. In EI the value lies between -1 (no similarity) to +1 (equal). To present the results of all the metrics in the same graph we normalize the values of EI metric between 0 to 1.

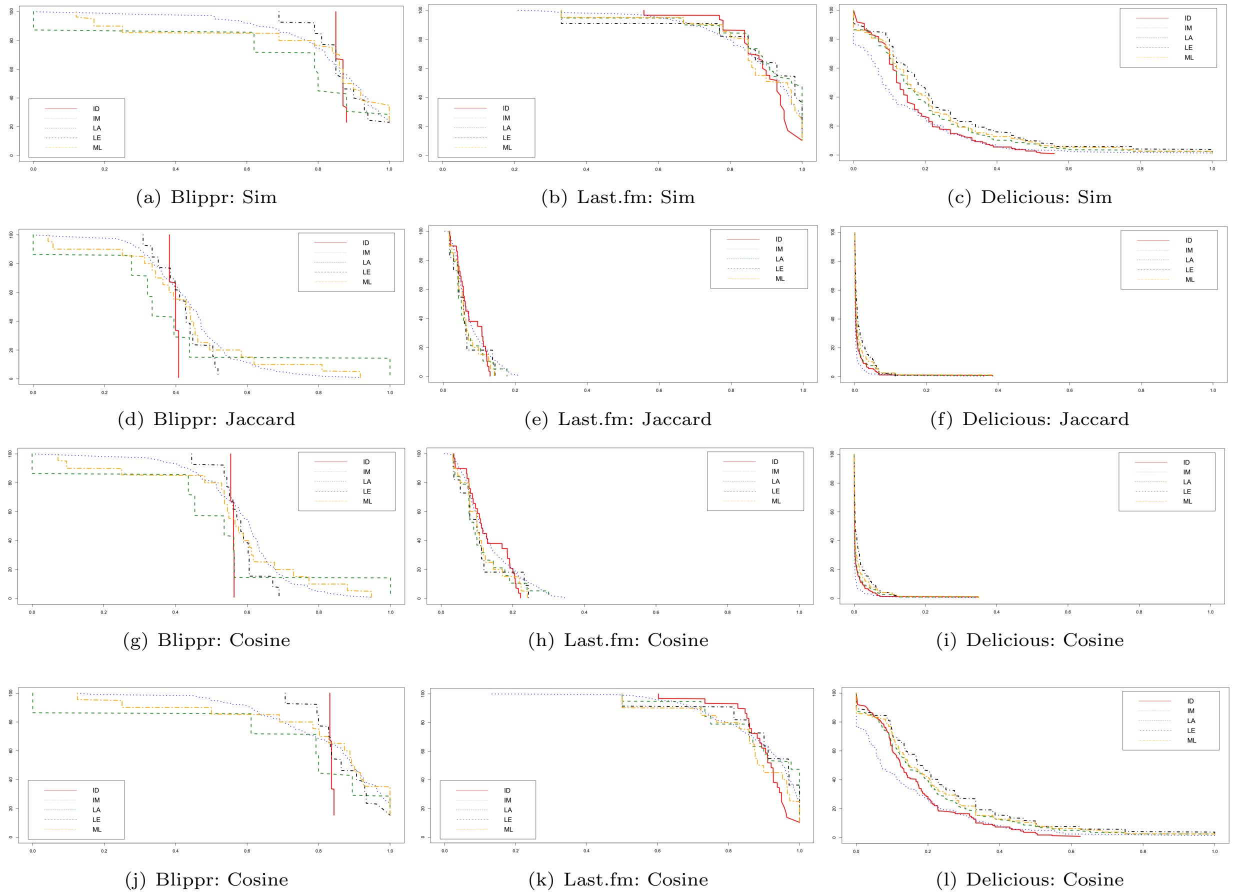
3.3 Results

We now describe various results of various similarity measures on the three datasets.

3.3.1 **Similarity measures**. For each community obtained through various community detection algorithms, we measure the similarity values for each of the similarity metric (Sim, Jaccard, Cosine and EI) in each dataset. Figure 2, shows the results of similarity values in the reverse cumulative frequency distribution. The X-axis represents the similarity values from 0 to 1 and Y-axis the % of communities. For each of the datasets and for each similarity metric, we plot the outcome of various community detection algorithms that is Demon (ID), Multilevel (ML), Leading Eigenvector (LE), Label Propagation (LA), Infomap (IM). As each of the algorithms produces a different number (and size) of communities, thus, we first normalize different similarity outcomes from various different community detection algorithms before plotting the values in the same graph.

As mentioned each graph shows the reverse cumulative distribution of the output. For example, in Figure 2(a), 30% of the communities have similarity value of 1. In case of Demon, 100 % of the communities have similarity value of at least 0.85. Thus, in general, if one expects that nodes within communities also have similar attributes, then the graphs should be more on the top right part of the box. That is more % of communities (top part of Y-axis) should have a high value of similarity (right part of the X-axis). The best similarity value is observed in the case of Blippr and Last.fm in case of *Sim* metric. Among the three datasets, Delicious shows the worst results for the similarity metrics. Among all the metrics, Cosine and Jaccard show the worst values for the similarities.

The low similarity among the nodes' attributes in a social network could be due to various reasons. Sometimes individuals become friends on online social platforms such as Facebook because of a meeting at a social gathering. In



(a) Blippr: Sim (b) Last.fm: Sim (c) Delicious: Sim

(d) Blippr: Jaccard (e) Last.fm: Jaccard (f) Delicious: Jaccard

(g) Blippr: Cosine (h) Last.fm: Cosine (i) Delicious: Cosine

(j) Blippr: Cosine (k) Last.fm: Cosine (l) Delicious: Cosine

Figure 2: Similarity Measurements.

another real-life example, generally we are friends with our neighbors, however, that does not mean we have common interests. Thus, it is not important that structure-based communities also exhibit strong nodes similarity in terms of attributes.

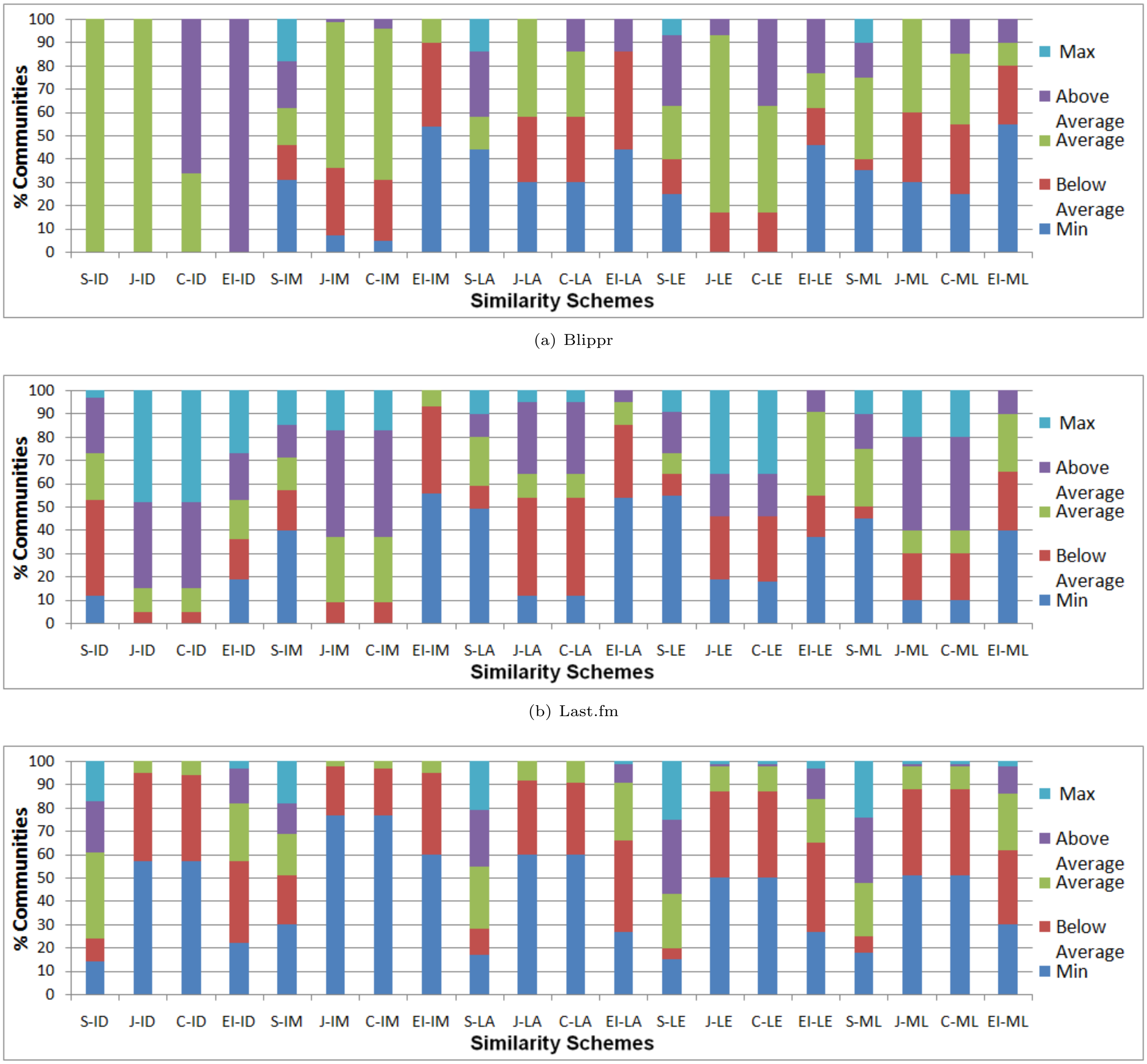
3.3.2 **Entropy**. Within a community to measure the difference among the similarity output, we also calculate the entropy for all the four metrics using following standard formulation:

$$H(x) = -\sum_{j=1}^{C} S(c_j) log_2 S(c_j) \tag{1}$$

Where H(x) represents the final outcome of entropy, which is calculated by summing up the individual similarity value for each pair of nodes and among all the communities (j=1 to C), using a similarity function, represented by S. Lower values of the entropy mean there is not much difference among the values of similarities within a community. Figure 3 shows the output for all the three datasets. For each community formed using various community detection algorithms, that is, Demon (ID), Multilevel (ML), Leading Eigenvector (LE), Label Propagation (LA), Infomap (IM), we aggregate all the similarity values that is Sim (Sim), Jaccard (Jac), Cosine (Cos) and EI.

We normalized the values of similarity between 0 to 1 for all the similarity metrics. For simplicity, we categorise the entropy outcomes in five categories namely i) Min (representing 0 <= H(x) < 0.2 for Sim and 0 <= H(x) < 1 for others), ii) below average (representing 0.2 <= H(x) < 0.4 for Sim and 1 <= H(x) < 2 for others), iii) average (representing 0.4 <= H(x) < 0.6 for Sim and 2 <= H(x) < 3 for others), iv) above average (representing 0.6 <= H(x) < 0.8 for Sim and 3 <= H(x) < 4 for others), and v) max (representing 0.8 <= H(x) <= 1 for Sim and 4 <= H(x) for others).

A high entropy value means the similarity values within a community varies a lot among various neighboring nodes compared when the entropy is low. In Blippr (Figure 3(a)), except in case of Cosine (c) and EI (EI) metric for communities detected by Demon (D) algorithm, in all other combinations, at least 60 % of communities have less than or equal to average entropy. Among all the three datasets, Last.fm shows worst results as very few communities have a low range of entropies. Delicious shows the lowest entropy, with most of the communities, are less than *Average* category. Low entropy for Delicious also means that low similarity metrics



(a) Blippr

(b) Last.fm

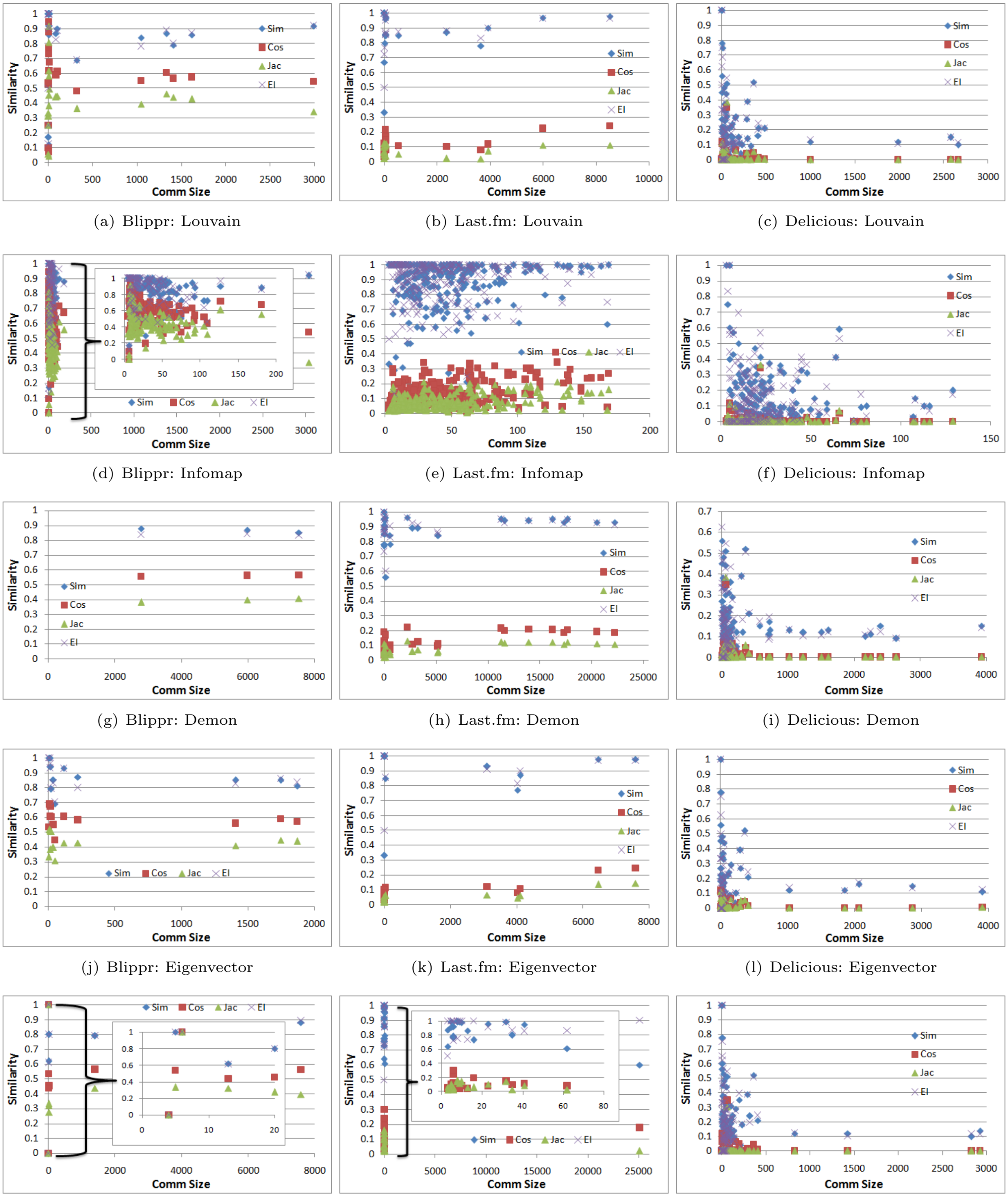
(c) Delicious

Figure 3: Entropy.

(see sub Figures (c), (i), (f), (l) in Figure 3) is consistently present among various communities. In other words, most of the nodes in the Delicious communities are not similar in terms of their corresponding interests.

3.3.3 **Does size matters?** We also analysed if the size of communities has any impact on the similarity output. The hypothesis behind this analysis is due to the fact that in small communities, people tend to have similar interests and in bigger communities, there is a high probability of varying interests being present among the nodes of a community.

Figure 4 shows the outcome of our analysis. For each dataset and for each community, we plot the values between Community Size (X-axis) and similarity values (Y-axis) for each of the metrics, that is Sim, Cosine (Cos), Jaccard (Jac), EI. One common observation from all the figures is that *Sim* and EI show more similar behavior and on the other side, Jaccard and Cosine return similar similarity values. In case of Last.fm and Blippr, larger communities have always shown better values for Sim and EI metrics. However, this is not observed in the case of the Delicious dataset.



(a) Blippr: Louvain

(b) Last.fm: Louvain

(c) Delicious: Louvain

(d) Blippr: Infomap

(e) Last.fm: Infomap

(f) Delicious: Infomap

(g) Blippr: Demon

(h) Last.fm: Demon

(i) Delicious: Demon

(j) Blippr: Eigenvector

(k) Last.fm: Eigenvector

(l) Delicious: Eigenvector

(m) Blippr: Label Prop

(n) Last.fm: Label Prop

(o) Delicious: Label Prop

Figure 4: Community Size Vs. Similarities.

For each community, we also measured the correlation between the size of the community and the similarity metric. Table 3 presents the results. *Sim* and EI again shows the direction similarity for all the datasets (except in the case of

Algo.	Blippr				Delicious				Last.fm			
	Sim	Jaccard	Cosine	EI	Sim	Jaccard	Cosine	EI	Sim	Jaccard	Cosine	EI
Louvain	0.1224	-0.0784	0.0112	0.1069	-0.1089	-0.0729	-0.0812	-0.1119	0.1438	0.1475	0.5910	0.1316
Infomap	0.0277	-0.0795	-0.0398	0.0291	-0.0700	-0.0278	-0.0341	-0.0833	0.1170	0.3524	0.3477	0.1081
Demon	-0.9262	0.9999	0.98602	-0.1802	-0.0692	-0.0653	-0.0677	-0.0916	0.2849	0.7195	0.7053	0.2102
Eigenvector	-0.1549	-0.1941	-0.2040	-0.0700	-0.1424	-0.1326	-0.1383	-0.1442	0.1752	0.8502	0.8509	0.1446
Label Prop	0.2123	0.0113	0.1003	0.2285	-0.0663	-0.0180	-0.0408	-0.0778	-0.4569	-0.2640	-0.2307	0.3215

Table 3: Correlation values between community size and similarity values.

label propagation algorithm for the Delicious dataset). Only 10% of the values are 70 % or more correlated which is not a significant result. Thus, there is no clear correlation that exists between the community size and the similarity metric.

4 CONCLUSION AND FUTURE WORK

In this paper, we investigated if the nodes in the communities derived from topological based community detection algorithms also exhibit interest similarity. We performed our analysis using three real social network datasets and five topological based community detection algorithm. In contrast to the general belief that *bird of the same feather flock together*, our results portray a different picture. We found out that on an average among all the communities, around 50 % of the nodes have similar interests.

We plan to extend this work by analysing larger social network datasets to confirm our understanding. We would also like to incorporate more 1) metrics of similarities and 2) community detection algorithms in our analysis. We also would like to give a more statistical treatment to our analysis for our future work.

5 ACKNOWLEDGMENTS

This work has been supported in part by H2020 framework project, SoBigData, grant number 654024.

REFERENCES

[1] 2012. In *Advanced Data Mining and Applications*. Lecture Notes in Computer Science, Vol. 7713.

[2] V.D. Blondel, J.L. Guillaume, R. Lambiotte, and E.L.J.S. Mech. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* (2008), P10008.

[3] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. 2011. 2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011). In *Proceedings of the 5th ACM conference on Recommender systems (RecSys 2011)*. ACM, New York, NY, USA.

[4] Aaron Clauset, Cristopher Moore, and M. E. J. Newman. 2004. Finding community structure in very large networks. *Physical Review* 70 (2004).

[5] Denzil Correa, Ashish Sureka, and Mayank Pundir. 2012. iTop: Interaction Based Topic Centric Community Discovery on Twitter. In *Proceedings of the 5th Ph.D. Workshop on Information and Knowledge (PIKM '12)*. ACM, New York, NY, USA, 51–58.

[6] Michele Coscia, Fosca Giannotti, and Dino Pedreschi. 2011. A Classification for Community Discovery Methods in Complex Networks. *Stat. Anal. Data Min.* 4, 5 (Oct. 2011), 512–546.

[7] Michele Coscia, Giulio Rossetti, Fosca Giannotti, and Dino Pedreschi. 2014. Uncovering Hierarchical and Overlapping Communities with a Local-First Approach. *ACM Trans. Knowl. Discov. Data* 9, 1, Article 6 (Aug. 2014), 27 pages.

[8] Santo Fortunato. 2010. Community detection in graphs. *Physics Reports* 486, 3-5 (2010), 75 – 174.

[9] Esther Galbrun, Aristides Gionis, and Nikolaj Tatti. 2014. Overlapping community detection in labeled graphs. *Data Mining and Knowledge Discovery* 28, 5 (2014), 1586–1610.

[10] Robert A. Hanneman and Mark Riddle. 2005. *Introduction to social network methods*. University of California, Riverside.

[11] Itai Himelboim, Stephen McCreery, and Marc Smith. 2013. Birds of a Feather Tweet Together: Integrating Network and Content Analyses to Examine Cross-Ideology Exposure on Twitter. *Journal of Computer-Mediated Communication* 18, 2 (2013), 40–60.

[12] Darko Hric, Richard K. Darst, and Santo Fortunato. 2014. Community detection in networks: Structural communities versus ground truth. *Phys. Rev. E* 90 (Dec 2014), 062805. Issue 6.

[13] Malek Jebabli, Hocine Cherifi, Chantal Cherifi, and Atef Hamouda. 2018. Community detection algorithm evaluation with ground-truth data. *Physica A: Statistical Mechanics and its Applications* 492 (2018), 651 – 706.

[14] Kwan Hui Lim and Amitava Datta. 2013. A Topological Approach for Detecting Twitter Communities with Common Interests., Vol. 8329. Ubiquitous Social Media Analysis., 23–43.

[15] 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.

[16] M. E. J. Newman. 2006. Finding community structure in networks using the eigenvectors of matrices. *Physical review E* 74, 3 (2006). cite arxiv:physics/0605087Comment: 22 pages, 8 figures, minor corrections in this version.

[17] Mark E. J. Newman and Aaron Clauset. 2015. Structure and inference in annotated networks. *CoRR* abs/1507.04001 (2015).

[18] Diana Palsetia, Md. Mostofa Ali Patwary, Kunpeng Zhang, Kathy Lee, Christopher Moran, Yves Xie, Daniel Honbo, Ankit Agrawal, Wei keng Liao, and Alok Choudhary. 2012. User-Interest based Community Extraction in Social Networks *(The 6th SNA-KDD Workshop)*.

[19] Leto Peel, Daniel B. Larremore, and Aaron Clauset. 2017. The ground truth about metadata and community detection in networks. *Science Advances* 3, 5 (2017). https://doi.org/10.1126/sciadv.1602548

[20] U. N. Raghavan, R. Albert, and S. Kumara. 2007. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E* 76, 3 (2007).

[21] Martin Rosvall and Carl T. Bergstrom. 2008. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences* 105, 4 (2008), 1118–1123.

[22] Rajesh Sharma, Matteo Magnani, and Danilo Montesi. 2015. Understanding Community Patterns in Large Attributed Social Networks. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 (ASONAM '15)*. ACM, New York, NY, USA, 1503–1508.

[23] Ken Wakita and Toshiyuki Tsurumi. 2007. Finding Community Structure in Mega-scale Social Networks: [Extended Abstract]. In *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*. ACM, New York, NY, USA, 1275–1276.

[24] Jierui Xie, Stephen Kelley, and Boleslaw K. Szymanski. 2013. Overlapping community detection in networks: The state-of-the-art and comparative study. *ACM Comput. Surv.* 45, 4 (2013), 43.

[25] Jaewon Yang and J. Leskovec. 2012. Community-Affiliation Graph Model for Overlapping Network Community Detection. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*. 1170–1175.