Community Detection Using Semilocal Topological Features and Label Propagation Algorithm

Deepanshu Malhotra[1], Ralucca Gera[2], and Akrati Saxena[3](✉)

[1] University School of Information, Communication and Technology, Guru Gobind Singh Indraprastha University, New Delhi, India
[2] Naval Postgraduate School, Monterey, CA 93943, USA
rgera@nps.edu
[3] Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands
a.saxena@tue.nl

Abstract. The detection of cohesive clusters with similar characteristics in multiple types of networks is of immense informational value to researchers. In this work, we propose a Weighted Semilocal Similarity based Label Propagation Algorithm (WSSLPA) for such community detection. The proposed method detects communities by using the semilocal topological features to overcome the shortcoming of randomness and instability in the existing label propagation algorithms while selecting a community label from multiple maximum labels. We associate user-defined weight parameters with the topological features to help WSSLPA adapt to different networks and enhance the performance metrics scores of detected communities. We compare the performance of the proposed method with other community detection techniques and show that the identified communities of WSSLPA are closer to the ground-truth communities.

Keywords: Complex networks · Community detection · Label propagation

1 Introduction

The emergence of network science has put forth the complex networks that model the intricate relationships among the components of various complex systems [1]. The inception of such complex networks attract researchers' interest to various problems, such as their evolution [25], identify influential nodes [26], link prediction [17], information diffusion [9], and so on. Community detection is one of the fundamental problems in Network Science that aims to find strongly connected clusters of nodes in a network, identified through more intra-cluster edges than inter-cluster edges. A plethora of algorithms exists that are driven by different motivations behind finding the clusters. The well known approaches are based

D. Mohaisen and R. Jin (Eds.): CSoNet 2021, LNCS 13116, pp. 255–266, 2021.
https://doi.org/10.1007/978-3-030-91434-9_23

on modularity optimization [19], information-theoretic techniques [10], genetic algorithms [16], non-negative matrix factorization [3], and label propagation [7].

One algorithm that we build our work on is by Raghavan et al. [22], who proposed the Label Propagation Algorithm (LPA) that utilizes the neighborhood structures of nodes to detect communities in networks. This approach consists of three prominent steps., In the initialization phase, nodes are randomly assigned a unique community label. Following that, the label propagation step begins wherein each node adapts a label that is assigned to the majority of its neighbors. Finally, the algorithm terminates when nodes have a label that belongs to the majority of their neighbors. The nodes having the same label get clustered together, thus producing the community structure of a given network.

The time efficiency and simplicity are advantages of LPA that have encouraged the development of various new approaches in this direction. Poaka et al. [20] developed a new LPA approach wherein they compute link density based measures to avoid ties between multiple maximum labels. They also extended their method by using fuzzy techniques to find overlapping communities in complex networks. Jokar et al. [11] extended the previous approach by developing a new metric that utilized link density to choose the future community of a node in the case of multiple maximum labels; and also presented a balancing parameter that assigned appropriate weights to the similarity measures between the node pairs. Verma et al. [28] developed a semi-supervised learning technique based on LPA to find communities in complex networks; the proposed method initializes the communities using core nodes identified through various centrality measures. Li et al. [14] developed an improved LPA by utilizing the modularity function and node importance, i.e., the normalized degree centrality of the node. The proposed algorithm first initializes the communities using the modularity function and then performs the update step in a specific order by using the node properties.

In light of the context above, we propose a label propagation based community detection method to find better quality communities. We will address two shortcomings [7] of the existing algorithms, (i) the flaw of randomness and (ii) the lack of stability in LPA; that are encountered when the algorithm randomly selects a future community for each node in case of multiple maximum labels. Therefore, LPA is unable to achieve stable community structures that can be observed by its lower metric scores that are computed by taking the mean value on several runs. Our work addresses these issues by utilizing topology based similarity measures, thereby breaking the tie of random assignment from multiple maximum labels in obtaining a community structure. Furthermore, the proposed method produces a stable community structure that is demonstrated by high metric scores obtained on various networks.

In complex networks, the global similarity measures consider the properties of the whole network that lead to high time complexity. On the other hand, local similarity measures examine only the immediate neighborhood information of each node. Henceforth, we use semilocal similarity measures to strike a balance between efficiency and information quantity. Among the semilocal indices, we use the extended Jaccard [2], and 2-hop neighborhood volume [29] to find suitable future communities for the nodes. Our proposed method also integrates user-defined parameters for the above-mentioned measures to better adapt to different networks. We conduct experiments comparing against other existing community detection methods on real-world networks and evaluate the identified communities using different performance metrics.

The rest of the paper is organized as follows. Section 2 describes the algorithm proposed in this study. Section 3 presents the experimental results for the comparison of our method with various existing community detection methods. Lastly, Sect. 4 holds the conclusion of this paper along with future directions.

2 Proposed Method

In this section, we present the details of our proposed method, namely the Weighted Semilocal Similarity based LPA (WSSLPA). As mentioned, the WSSLPA removes the random selection of the future community for a node in case of multiple maximum labels. This is done with the help of topological information that is fine-tuned by parameters.

We first discuss the parameters required for our proposed method. The Extended Jaccard EJ coefficient for two adjacent nodes u and v is defined as,

$$EJ_{(u,v)} = \frac{|\Gamma_2(u) \cap \Gamma_2(v)|}{|\Gamma_2(u) \cup \Gamma_2(v)|}, \tag{1}$$

where $\Gamma_2(u)$ denotes the union of the neighbors of a given node u that are either at one-hop or two-hop distance away from node u.

The 2-Hop Neighborhood volume NV^2 for a node u is defined as,

$$NV_u^2 = \sum_{w \in \Gamma_2(u)} deg(w), \tag{2}$$

where $deg(w)$ denotes the degree of node w and $\Gamma_2(u)$ is defined as in Eq. 1. Additionally, the similarity of a pair of nodes (u, v) is represented by $Sim_{u,v}$ that denotes the weighted sum of $EJ_{(u,v)}$ and NV_v^2 using Eq. 3.

$$Sim_{(u,v)} = k_1 \cdot EJ_{(u,v)} + k_2 \cdot \widetilde{NV_v^2} \tag{3}$$

where $\widetilde{NV_v^2}$ is the normalized value of NV_v^2. The NV^2 values obtained represent the sum of degrees, therefore we normalize the NV^2 values of all the nodes in the network to accommodate them in the interval of $[0, 1]$.

The steps of the WSSLPA algorithm are as follows.

1. WSSLPA takes a network $G(V, E)$ as an input, where V is the set of nodes and E is the set of edges in network G. The initialization phase begins, wherein WSSLPA assigns a unique community label to each node present in the network.

2. The label propagation phase begins. First, the algorithm arranges the nodes in random order. Next, it chooses a maximum community label in the neighborhood of each node and assigns it as the node's future community.
3. In the case of multiple maximum labels being assigned to a single node, say u, we use the Extended Jaccard (EJ) introduced in Eq. 1, and 2-Hop Neighborhood Volume $(\widetilde{NV^2})$ introduced in Eq. 2 as follows.
   (a) Both EJ and $\widetilde{NV^2}$ are pre-computed for the fast execution of the proposed method for all the edges and nodes, respectively.
   (b) Next, $Sim_{(u,v)}$ is computed for every pair of nodes using EJ, $\widetilde{NV^2}$ and the user-defined parameters as explained in Eq. 3.
4. We compute a community wise cumulative sum of the combined similarity measure, that is $Sim^{c_i}_{sum}(u) = \sum_{CommunityLabel(v)=c_i \ \& \ (u,v)\in E} Sim_{(u,v)}$, $\forall c_i \in C$, where $C = \{c_1, c_2, \cdots, c_i, \cdots\}$ is the set of community labels. Subsequently, the community label with maximum sum magnitude is selected to be the future community of the given node, therefore, $CommunityLabel(u) = argmax_{c_i}\{Sim^{c_i}_{sum}(u), \forall c_i \in C\}$.
5. The algorithm is terminated if all the nodes have a label that belongs to the majority of their neighbors or the maximum number of iterations (t) is reached.

We now delineate the complete steps for WSSLPA in Algorithm 1.

Algorithm 1: Weighted semilocal similarity based LPA

Input: $G(V, E)$: The Input Network, t: Maximum Iteration Limit

For each node u, assign a unique community label $CommunityLabel(u)$
$iterations \leftarrow 0$
repeat
  $V' \leftarrow$ Shuffle the list of nodes V to produce a random order
  for u *in* V' **do**
    if *Multiple maximum labels for node* u **then**
      Compute Sim score using Equation 3
      Calculate the community wise cumulative sum of Sim score, $Sim^{c_i}_{sum}(u) = \sum_{CommunityLabel(v)=c_i \ \& \ (u,v)\in E} Sim_{(u,v)}$, $\forall c_i \in C$
      $CommunityLabel(u) = argmax_{c_i}\{Sim^{c_i}_{sum}(u), \forall c_i \in C\}$
    else
      $CommunityLabel(u) \leftarrow$ Maximum label among the neighbors of node u
  end
until *All nodes have a label equal to the majority of their neighbors or* $iterations > t$
return $CommunityLabel$

2.1 Time Complexity

In this section, we present the time complexity of the proposed method. Let n denotes the number of nodes in the network, k_{avg} is the average degree for the nodes, t is the maximum number of iterations if the termination criteria is not satisfied (namely that of all nodes have a label matching most of their neighbors' label).

The first step of the proposed algorithm includes the calculation of semilocal measures. The time complexity for calculating these structural measures is equal to $O(nk_{avg}^2)$. Next, the algorithm begins with the initialization phase, which takes $O(n)$ time. Subsequently, the label propagation step of the proposed method is executed that has $O(tnk_{avg})$ time complexity. Finally, in the termination step, every node's neighborhood is utilized to check the termination criteria. This step has $O(nk_{avg})$ complexity. Henceforth, the overall time complexity of the proposed technique is $O(nk_{avg}^2 + tnk_{avg})$.

3 Experimental Analysis

In this section, we introduce the real-world and synthetic datasets used in the experiments. We then follow it with the performance analysis of the WSSLPA algorithm as compared to the baseline community detection algorithms.

3.1 Datasets

To evaluate the performance of the proposed algorithm, we use various real-world datasets, including Karate, Dolphins, Polbooks, Football, Cora, Citeseer, and AS internet network. The availability of ground-truth community structure is a critical requirement in our experiments, and thus we consider datasets having predefined ground-truth structures. Furthermore, we also test the algorithms on *LFR* benchmark network. In LFR, the minimum degree and minimum community size was set to 20, and the maximum degree and maximum community size was set to 50. Table 1 summarizes metrics of these networks.

3.2 Experimental Settings

For the analysis, we run each algorithm (WSSLPA and baselines) 10 times and report their averages when comparing against WSSLPA. The performance metrics we use to measure the overall quality of the community structures are the Normalized mutual information (NMI) [5], and modularity [19]. We use the termination criteria to set up a maximum number of iterations our proposed method can execute. This is applied if some nodes do not have a label that belongs to the majority of their neighbors, and the maximum number of iterations is set to 1000 for the majority of networks (the exception is *AS Internet* network, where we set this number to 100 as it is a large network).

The weight parameters, namely k_1 and k_2, constitute an important part of experimental settings for WSSLPA. They help the proposed method adapt to

Table 1. Description of datasets used in this study.

Dataset	Acronym	Nodes	Edges	#Ground-truth communities	Ref
Karate	Kar	34	78	2	[30]
Dolphins	Dol	62	159	2	[15]
Polbooks	Pol	105	441	3	[12]
Football	Foot	115	613	12	[8]
Cora	Cora	2708	5278	7	[27]
Citeseer	Cite	3327	4676	7	[27]
AS Internet	AS	23752	58416	176	[4]
LFR	LFR	500	$\mu(0.1 - 0.9)$	20–50	[13]

different networks efficiently and maintain a fine balance between the extended Jaccard and 2-hop neighborhood volume. The default value of weight parameters is set as $k_1 = 0.8$ and $k_2 = 0.2$ for WSSLPA as experimentation says that these settings provide better results compared to baselines for most of the datasets. Table 2 presents parameter values for all datasets that provides best results (shown in Table 3) based on the experimental observation.

Table 2. Parameters value (k_1, k_2) for the best results of WSSLPA.

Datasets	Karate	Dolphins	Polbooks	Football	Cora	Citeseer
k_1, k_2	0.9, 0.2	0.2, 0.8	0.9, 0.1	0.9, 0.2	0.1, 1.0	1.0, 0.1
Datasets	AS	LFR($\mu = 0.1$)	LFR($\mu = 0.3$)	LFR($\mu = 0.5$)	LFR($\mu = 0.7$)	LFR($\mu = 0.9$)
k_1, k_2	0.9, 0.2	1.0, 0.1	0.2, 0.1	0.7, 0.5	0.5, 0.5	0.5, 0.5

3.3 Performance Analysis

Table 3 presents the performance comparison of WSSLPA with four community detection techniques. We compare against four established methods: leading eigenvector algorithm (Lead) [18], LPA [22], walktrap (Walk) [21], and infomap (Info) [23,24] algorithms. For the WSSLPA method we show both the best results achieved, as well as the results for the default parameter setting.

We observe that WSSLPA achieves the best NMI scores on the majority of networks (0.8209, 0.8483, & 0.3227 are the best NMI scores achieved by WSSLPA on *Kar, Dol, Cite*, respectively), and competitive scores on others (0.5619, 0.9150, & 0.3342 are the second best NMI scores obtained on *Pol, Foot, AS*, respectively). WSSLPA with default parameters also obtains better NMI results (on *Kar, Cite, AS*) in comparison with other community detection methods. Infomap and LPA achieve the best NMI scores on some networks; however, WSSLPA gives competitive results on them. Additionally, the Lead and Walk

Table 3. Performance comparison using the NMI and modularity metrics

Data Set	Algorithm	Lead	LPA	Walk	Info	WSSLPA (best)	WSSLPA (0.8/0.2)
Kar	NMI	0.6771	0.6815	0.6110	0.6994	0.8209	0.7901
	Modularity	0.3934	0.3604	0.3431	0.4020	0.3970	0.3925
Dol	NMI	0.4489	0.6377	0.5372	0.5844	0.8483	0.5761
	Modularity	0.4911	0.4795	0.4888	0.5269	0.4331	0.4955
Pol	NMI	0.5201	0.5655	0.5081	0.4934	0.5619	0.5483
	Modularity	0.4671	0.4989	0.4961	0.5228	0.4904	0.4893
Foot	NMI	0.6986	0.8679	0.7451	0.9241	0.9150	0.9006
	Modularity	0.4926	0.5871	0.5883	0.6005	0.5807	0.5739
Cora	NMI	0.3820	0.4233	0.4011	0.4128	0.4111	0.4002
	Modularity	0.7318	0.7401	0.5888	0.7178	0.7339	0.6237
Cite	NMI	0.3011	0.3114	0.3181	0.3119	0.3227	0.3223
	Modularity	0.8541	0.8221	0.8089	0.8207	0.7531	0.7534
AS	NMI	0.0000	0.2302	0.2553	0.4412	0.3342	0.3092
	Modularity	0.0000	0.2010	0.1649	0.5195	0.3540	0.3139

algorithms obtain low NMI scores. For modularity analysis, WSSLPA performs competitively in most cases while it gets low modularity scores on some networks.

The main aim of this study was to develop a community detection technique that could produce near ground-truth community structures. The proposed WSSLPA method performs excellently on the NMI measure as it produces high quality community structures. The modularity community score suffers from the resolution limit problem wherein it rewards the large size communities while ignoring the small communities [6]. This is one of the main reasons for the average performance of WSSLPA on the modularity metric. Overall, WSSLPA produces better performances across various networks as observed in Table 3 summarizing the results of our experiments.

Additionally, we perform an experiment to evaluate different community detection methods utilized in this study on the LFR benchmark datasets, running each experiment 10 times and showing the average values. We create the LFR network with 500 nodes by varying the mixing parameter $\mu \in [0.1, 0.9]$. Figure 1 presents the results for this experiment, where the x-axis denotes the different mixing parameter values, while the y-axis represents the NMI scores obtained by different algorithms.

We observe that the infomap method obtains the highest NMI score for smaller μ (namely $\mu \in \{0.1, 0.3\}$), but its NMI values fall sharply after that to their lowest point. Similar results are observed for LPA wherein the NMI values fall sharply after the mixing parameter value of 0.1. WSSLPA achieves the highest score at $\mu = 0.1$, and it drops as observed for other methods. However, the WSSLPA achieves the highest NMI values in comparison to other algorithms

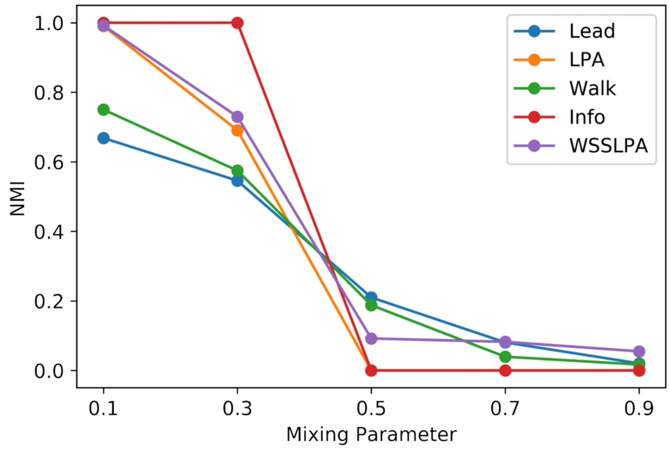


Fig. 1. Comparison on the LFR network using the NMI performance metric.

for $\mu \in \{0.7, 0.9\}$. This observation shows the robust nature of the proposed community detection technique for different mixing parameter values. Lead and Walktrap methods achieve low NMI scores in the majority of the cases except at $\mu = 0.5$, where they obtain higher NMI values compared to other algorithms.

The above experiments help in providing a deep insight into the performances of various community detection techniques. They also help in exhibiting the consistent performances of WSSLPA across a variety of datasets with the help of different evaluation metrics. WSSLPA obtains higher NMI scores on real-world and synthetic networks, signifying the superior quality of identified communities.

3.4 Sensitivity Analysis

We now study the impact of weight parameters, i.e., k_1, k_2, on the performance of WSSLPA for different networks. For this analysis, the one parameter (k_1 or k_2) will be set to 0.5 and the other will be varied in the range of $[0.1, 0.9]$. Figures 2, and 3 show the NMI values of identified communities. In Fig. 2, $k_2 = 0.5$, while $k_1 \in [0.1, 0.9]$, and in Fig. 3, $k_1 = 0.5$, while k_2 varies in the range $[0.1, 0.9]$.

We observe from Fig. 2 that the overall NMI scores of WSSLPA increase as the value of k_1 increases. In Fig. 3, we observe that the overall NMI scores decrease with the increasing value of k_2. The variation in k_2 affects more the performance on small size networks as compared to larger networks.

In Fig. 4, we show the NMI values obtained by WSSLPA on *Kar, Dol, Pol, Foot* networks for all parameter (k_1 and k_2) settings. The results show that a higher value of k_1 and a lower value of k_2 provide good results on most networks, as expected.

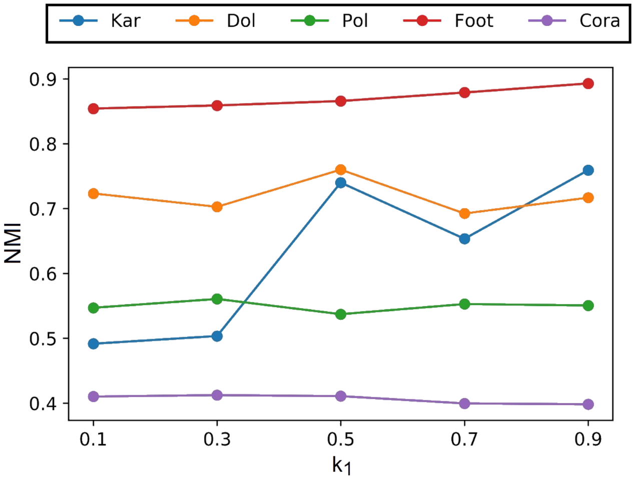


Fig. 2. Analysis of WSSLPA by varying $k_1 \in [0.1 - 0.9]$ while keeping $k_2 = 0.5$.

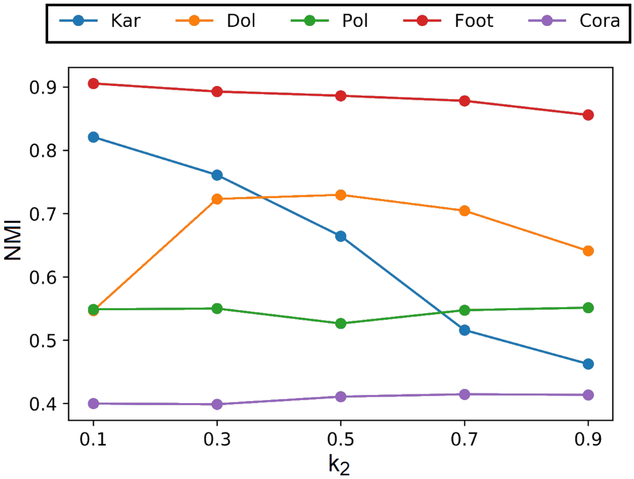
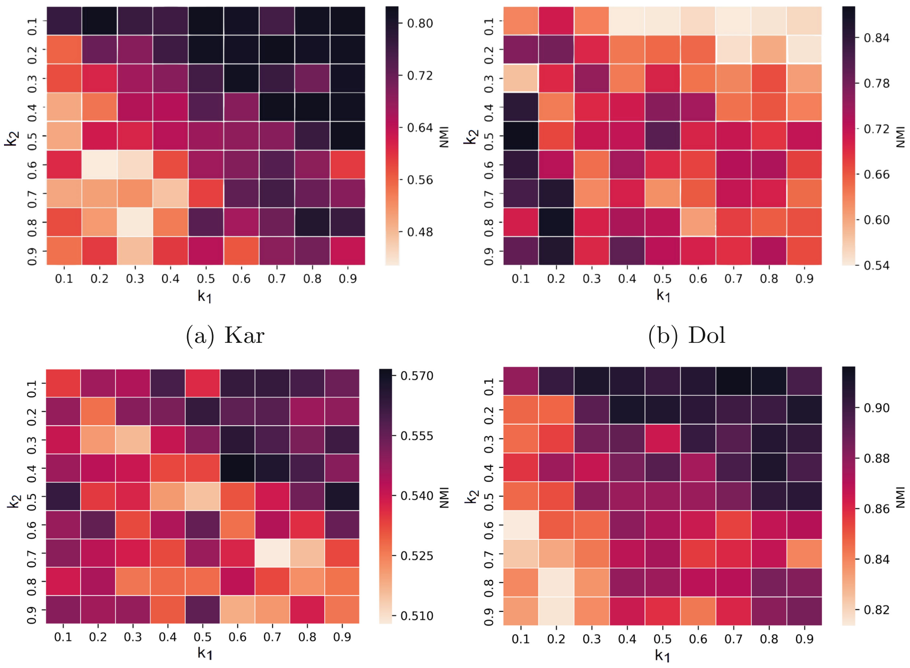


Fig. 3. Analysis of WSSLPA by varying $k_2 \in [0.1 - 0.9]$, and $k_1 = 0.5$.

We, therefore, conclude that on the majority of networks, we obtain better NMI scores through the combination of higher k_1 values and lower k_2 values. An exception is the case of *Cora* network wherein lower k_1 and higher k_2 values give better score. This might be because, in these networks, a higher preference is given to the centrally connected nodes while predicting clusters closer to ground-truth community structures. Henceforth, the parameters for WSSLPA are based on these observations that finally help the proposed algorithm achieve better quality community structures.



(a) Kar (b) Dol

(c) Pol (d) Foot

Fig. 4. Analysis of WSSLPA on *Kar, Dol, Pol*, & *Foot* networks for $k_1 \in [0.1 - 0.9]$ and $k_2 \in [0.1 - 0.9]$.

4 Conclusion

The development of efficient and accurate community detection methods is a keen research area in the field of network science. Recent methods propose the utilization of local information to detect the densely connected communities of nodes. Although such methods are efficient, they achieve low performances because of the limited information extracted from the network. Furthermore, global similarity methods consider the topology of the whole network, thereby making them less efficient.

In this study, we presented a Label Propagation Algorithm (LPA), named Weighted Semilocal Similarity based Label Propagation Algorithm (WSSLPA), that detects high quality communities that are similar to the ground-truth communities in complex networks. Our proposed method utilized semilocal similarity measures to counter the shortcoming of randomness in LPA. Consequently, this improved the quality of WSSLPA's detected communities by avoiding the formation of large communities, which is a shortcoming in most modularity-based community detection methods. Additionally, the utilization of semilocal measures helped in retrieving considerable global network information, while the label propagation technique assisted in improving the algorithm run time. The experimental results showed the better performance of WSSLPA on real-world as well as on synthetic networks as compared to baseline methods. The proposed method performs consistently on different networks and achieves competitive NMI scores, thereby signifying the closeness of the identified communities with respect to the ground truth community structure.

One can further extend the proposed method to attributed networks where each node has different properties, and the edges might represent varied relationships. Such diverse information can be harnessed by developing efficient semilocal methods that can be utilized by the community detection algorithms. Furthermore, the existing method could be extended to detect overlapping communities wherein each node can attain multiple community labels. This would help to further develop novel community detection techniques for real-world systems where the objects are heterogeneous, and a single object might be linked with multiple communities.

References

1. Barabási, A.L.: Network science. Philos. Trans. R. Soc. Math. Phys. Eng. Sci. **371**(1987), 20120375 (2013)
2. Berahmand, K., Bouyer, A., Vasighi, M.: Community detection in complex networks by detecting and expanding core nodes through extended local similarity of nodes. IEEE Tran. Computat. Soc. Syst. **5**(4), 1021–1033 (2018)
3. Binesh, N., Rezghi, M.: Fuzzy clustering in community detection based on nonnegative matrix factorization with two novel evaluation criteria. Appl. Soft Comput. **69**, 689–703 (2018)
4. Boguná, M., Papadopoulos, F., Krioukov, D.: Sustaining the internet with hyperbolic mapping. Nat. Commun. **1**(1), 1–8 (2010)
5. Danon, L., Diaz-Guilera, A., Duch, J., Arenas, A.: Comparing community structure identification. J. Stat. Mech: Theor. Exp. **2005**(09), P09008 (2005)
6. Fortunato, S., Barthelemy, M.: Resolution limit in community detection. Proc. Natl. Acad. Sci. **104**(1), 36–41 (2007)
7. Garza, S.E., Schaeffer, S.E.: Community detection with the label propagation algorithm: a survey. Physica A **534**, 122058 (2019)
8. Girvan, M., Newman, M.E.: Community structure in social and biological networks. Proc. Natl. Acad. Sci. **99**(12), 7821–7826 (2002)
9. Guille, A., Hacid, H., Favre, C., Zighed, D.A.: Information diffusion in online social networks: a survey. ACM SIGMOD Rec. **42**(2), 17–28 (2013)
10. Hajek, B., Wu, Y., Xu, J.: Information limits for recovering a hidden community. IEEE Trans. Inf. Theor. **63**(8), 4729–4745 (2017)
11. Jokar, E., Mosleh, M.: Community detection in social networks based on improved label propagation algorithm and balanced link density. Phys. Lett. A **383**(8), 718–727 (2019)
12. Krebs, V.: Proxy networks. analyzing one network to reveal another. Bull. de Méthodol. Sociol. Bull. Sociol. Methodol. (79), 61–70 (2003)
13. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Phys. Rev. E **78**(4), 046110 (2008)
14. Li, H., Zhang, R., Zhao, Z., Liu, X.: LPA-MNI: an improved label propagation algorithm based on modularity and node importance for community detection. Entropy **23**(5), 497 (2021)

15. Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E., Dawson, S.M.: The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behav. Ecol. Sociobiol. **54**(4), 396–405 (2003)
16. Malhotra, D.: Community detection in complex networks using link strength-based hybrid genetic algorithm. SN Comput. Sci. **2**(1), 1–16 (2021)
17. Martínez, V., Berzal, F., Cubero, J.C.: A survey of link prediction in complex networks. ACM Comput. Surv. (CSUR) **49**(4), 1–33 (2016)
18. Newman, M.E.: Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E **74**(3), 036104 (2006)
19. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E **69**(2), 026113 (2004)
20. Poaka, V., Hartmann, S., Ma, H., Steinmetz, D.: A link-density-based algorithm for finding communities in social networks. In: Link, S., Trujillo, J.C. (eds.) Advances in Conceptual Modeling, ER 2016. LNCS, vol. 9975, pp. 76–85. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-47717-6_7
21. Pons, P., Latapy, M.: Computing communities in large networks using random walks. In: Yolum, p, Güngör, T., Gürgen, F., Özturan, C. (eds.) Computer and Information Sciences - ISCIS 2005, ISCIS 2005. LNCS, vol. 3733, pp. 284–293. Springer, Heidelberg (2005). https://doi.org/10.1007/11569596_31
22. Raghavan, U.N., Albert, R., Kumara, S.: Near linear time algorithm to detect community structures in large-scale networks. Phys. Rev. E **76**(3), 036106 (2007)
23. Rosvall, M., Axelsson, D., Bergstrom, C.T.: The map equation. Eur. Phys. J. Spec. Top. **178**(1), 13–23 (2009)
24. Rosvall, M., Bergstrom, C.T.: An information-theoretic framework for resolving community structure in complex networks. Proc. Natl. Acad. Sci. **104**(18), 7327–7331 (2007)
25. Saxena, A.: A survey of evolving models for weighted complex networks based on their dynamics and evolution. arXiv preprint arXiv:2012.08166 (2020)
26. Saxena, A., Iyengar, S.: Centrality measures in complex networks: A survey. arXiv preprint arXiv:2011.07190 (2020)
27. Sen, P., Namata, G., Bilgic, M., Getoor, L., Galligher, B., Eliassi-Rad, T.: Collective classification in network data. AI Mag. **29**(3), 93–93 (2008)
28. Verma, P., Goyal, R.: Influence propagation based community detection in complex networks. Machine Learning with Applications **3**, 100019 (2021)
29. Wehmuth, K., Ziviani, A.: Daccer: distributed assessment of the closeness centrality ranking in complex networks. Comput. Netw. **57**(13), 2536–2548 (2013)
30. Zachary, W.W.: An information flow model for conflict and fission in small groups. J. Anthropol. Res. **33**(4), 452–473 (1977)