Proc. 7th Workshop on Complex Networks (CompleNet), Dijon, France, Mar. 23-27, 2016,
Studies in Artificial Intelligence, **644**, Springer, Zurich, Switzerland, pp. 145-158.

Improving Network Community Structure with Link Prediction Ranking

Mingming Chen, Ashwin Bahulkar, Konstantin Kuzmin and Boleslaw K. Szymanski

Abstract Community detection is an important step of network analysis that relies on the correctness of edges. However, incompleteness and inaccuracy of network data collection methods often cause the communities based on the collected datasets to be different from the ground truth. In this paper, we aim to recover or improve the network community structure using scores provided by different link prediction techniques to replace a fraction of low ranking existing links with top ranked predicted links. Experimental results show that applying our approach to different networks can significantly refine community structure. We also show that predictions of edge additions and persistence are confirmed by the future states of evolving social networks. Another important finding is that not every metric performs equally well on all networks. We observe that performance of link prediction ranking is correlated with certain network properties, such as the network size or average node degree.

Keywords Community structure · Link ranking · Network dynamics

This work was supported in part by the ARL under Cooperative Agreement Number W911NF-09-2-0053 and by the Office of Naval Research Grants No. N00014-09-1-0607 and N00014-15-1-2640, and by the EU's 7FP Grant Agreement No. 316097 and by the Polish National Science Centre, the decision no. DEC-2013/09/B/ST6/02317.

M. Chen · A. Bahulkar · K. Kuzmin · B.K. Szymanski (✉)
Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
e-mail: szymab@rpi.edu

M. Chen
e-mail: chenm8@rpi.edu

A. Bahulkar
e-mail: bahula@rpi.edu

K. Kuzmin
e-mail: kuzmik@rpi.edu

B.K. Szymanski
Wroclaw University of Technology, 50-370 Wroclaw, Poland

H. Cherifi et al. (eds.), *Complex Networks VII*, Studies in Computational
Intelligence 644, DOI 10.1007/978-3-319-30569-1_11

1 Introduction

Detecting and characterizing network community structure are among the fundamental techniques of network science. Community detection reveals latent but meaningful structures in a wide range of networks [1], yet the results often do not represent the reality. The primary reason is that available network datasets are often incomplete or inaccurate because of lost, incorrect. or misrepresented data, especially when gathered from massive networks. Consequently, the networks derived from such data may have some edges missing while some edges present in the network dataset may not exist in reality.

In this paper we introduce and evaluate methods of recovering or improving the network community structure by removing extraneous (or transient) edges and restoring (or creating) the missing ones.

We start by setting the fraction of edges to be replaced which defines the number of added and deleted edges. Next, we rank all the edges by the chosen link prediction method. Then, we complement the network with non-existing highest ranked edges and remove the same number of existing lowest ranking links using three popular link prediction metrics. We evaluate this approach on seven real-world network datasets, including two friendship networks, two collaboration networks, and a co-purchasing network. After enhancing the networks with our link improvement procedure, we first run community detection algorithms to find community structure. Then we measure the quality of the discovered community structure with two global and six local metrics. The results show that the community structure of five out of seven real-world networks is significantly refined.

The rest of the paper is organized as follows. The related work is presented in Sect. 2. The detailed description of the approach is provided in Sect. 3. The analysis of the results using community structure metrics is given in Sect. 4. Section 5 includes the results for an evolving network with the known ground truth data. The conclusion and future work are outlined in Sect. 6.

2 Related Work

In [2] the authors focus on finding missing edges and communities. First, they classify the reasons for missing edges but do not consider extraneous edges. They use a partial network which is simulated by deleting a certain fraction of edges from the input dataset. The quality of the resulting communities is compared to their true versions which are assumed to be available. Normalized mutual information (NMI) [3] is used as a measure of community quality.

In [4] the authors use the results of community detection to guide the addition of missing links. In their approach, intra-community edges suggested by a link predictor are added to the network first, followed by inter-community edges. Experimental verification was performed on the LFR benchmark [5] and six very small real-world networks using several link predictors and two community detection algorithms.

A more detailed analysis of the relation between community structure and link formation is provided in [6]. Given an array of communities, the density of links inside a community and between any two communities determines the probability of adding a particular link. A further development [7] uses the network's local structural information for improved performance.

A common approach to enhancing the quality of community detection methods using link prediction techniques is to introduce a preprocessing step to ameliorate the network by reinforcing its community structure. An example is a method [8] in which link prediction is applied to assign weights to the existing edges of a network and then a community detection algorithm is applied to the weighted network. This approach uses five different community detection methods. The community quality is measured using NMI for synthetic networks and modularity for real-world ones.

A more complicated solution has been proposed recently in [9]. It involves running link prediction multiple times on the same input network thus creating a family of enhanced networks. Community detection is then performed for each network in this family. The final result is constructed by aggregating community detection results of each individual network. This approach was implemented only for disjoint community detection.

3 Link Replacing Methodology

Algorithm 1 defines our approach. First, every possible edge (whether existing or potential) in the network is assigned a rank based on the score returned by a link predictor $\mathscr{L}$. We use LPmade library [10] for unsupervised link prediction and analysis selecting three local computationally efficient metrics described below.

The Number of Common Neighbors (CN) is simply the count of the number of neighbors that any two given nodes have in common. The computational complexity to calculate this score for a network is $O(E)$, where E denotes the number of edges in the network.

Adamic-Adar (AA) [11] is a refinement of the CN in which each common neighbor of the two nodes for which the metric is evaluated adds the inverse of the logarithm of its degree to the result rather than adding a constant of 1. The computational complexity to calculate this score for a network is $O(E)$.

PropFlow (PF) [12] measures the geodesic proximity of the two nodes for which the metric is computed. We restrict the degree of the considered neighborhood to four, so the complexity of computing this metric is $O(d^4N)$ where d is the average node degree and N is the number of nodes in the network. For a sparse network, the complexity is linear in the number of nodes.

Edges and their corresponding rank scores are kept separately for existing and potential edges.

Algorithm 1 : Link ranking and replacement

```
Input: Graph $G = (V, E)$, link predictor $\mathscr{L}$, fraction of edges to be replaced $f$
Output: Graph $G' = (V, E')$ with improved community structure
  $E' \leftarrow E$
  $\overline{E} \leftarrow \{\{u, v\} : \forall u \in V, \forall v \in V, \text{ s.t. } u \neq v\} \setminus E$
  $R_E \leftarrow ()$
  $R_{\overline{E}} \leftarrow ()$
  for all $\{u, v\} \in E$ s.t. $(deg(u) > 1$ and $deg(v) > 1$ do
    Add $(\{u, v\}), \mathscr{L}(\{u, v\})$ to $R_E$
  end for
  Sort $R_E$ in the order of ascending rank values
  for all $e \in \overline{E}$ do
    Add $(e, \mathscr{L}(e))$ to $R_{\overline{E}}$
  end for
  Sort $R_{\overline{E}}$ in the order of descending rank values
  $n \leftarrow \lfloor f \cdot |E| \rfloor$
  for $i = 1$ to $n$ do
    $e \leftarrow$ edge from the $i^{th}$ top tuple of $R_{\overline{E}}$
    $E' \leftarrow E' \cup \{e\}$
  end for
  for $i = 1$ to $n$ do
    $e \leftarrow$ edge from the $i^{th}$ top tuple of $R_E$
    $E' \leftarrow E' \setminus \{e\}$
  end for
```

During the second phase of the algorithm, edge replacements take place. First, a number of edges (denoted by f) with the highest rank among the non-existing edges are added to the network. Next, the same number of the lowest ranked existing edges are removed from the network. In order to prevent the formation of isolated nodes (i.e., nodes with a degree of 0), an edge is not considered for removal if one or both of its endpoints have a degree of 1. Then, we use the community detection algorithms SpeakEasy [13] and GANXiS [14] to detect the community structure of the modified network.

SpeakEasy is a label propagation community detection algorithm which identifies communities using top-down and bottom-up approaches simultaneously. Specifically, nodes join communities based not only on the nodes' local connections but also on the global information about the network structure. It adopts consensus clustering to get robust community structure. In our experiments, we choose to make 50 label propagation iterations with no node receiving a new label before terminating. We conduct 20 replicate runs for consensus clustering to get more robust and deterministic results.

GANXiS is a fast algorithm using a general speaker-listener information propagation process. It spreads one label at a time between nodes according to the interaction rules. The worst-case time complexity of GANXiS is $O(E)$.

Both GANXiS and SpeakEasy can detect overlapping communities, but for our experiments we configure them to detect only disjoint communities. Once the community structure is found, we measure its quality using several metrics to check the performance of our approach. The impact of selecting the value of parameter f on performance is discussed in Sect. 4.3. Our approach differs from [2] since we do not know the ground truth networks. Instead, we consider different metrics (see Sect. 4.1 for details) and if the majority of them agree on the improvement of communities after the replacement, we accept the results.

4 Evaluation and Analysis

4.1 Community Quality Metrics

To evaluate our approach without ground truth, we adopt two global community quality metrics, modularity (Q) [15] and modularity density (Q_{ds}) [16], and the following six local community quality metrics: *Intra-density ID*, *Contraction CNT*, *Expansion EXP*, *Conductance CND* [16], *Fitness F* [17], and the *Modularity Degree D* [18]. For the sake of space, we list the metrics above, and refer the reader to the cited references for their formal definitions and descriptions.

4.2 Dataset Descriptions

We consider seven real-world network datasets, including two friendship networks, two collaboration networks, and a co-purchasing network. Below we describe the basic properties of these datasets and provide the number of nodes (N) and edges (E) for each.

Gowalla was collected from a location-based social networking provider. There are 391,222 users with public profiles (friends and check-ins) that were active from the middle of September 2011 to late October of the same year [19]. There are 2,176,188 edges in this network that indicate friendships between users.

Amazon is a product co-purchasing network of the Amazon website [20]. There are 334,864 nodes in the network that represent products and 925,872 edges that link commonly co-purchased products.

DBLP is a scientific collaboration network with 317,080 nodes representing authors and 1,049,866 edges connecting authors that have co-authored a paper [20].

Santa Fe is the largest connected component of the collaboration network of scientists at Santa Fe Institute during the years 1999 and 2000 [21]. It has 118 nodes and 200 edges.

Football is a network that represents the schedule of games between college football teams in a single season with 115 nodes and 613 edges [21].

Dolphin is a social network of frequent associations between 62 dolphins connected by 150 edges and living in a community off Doubtful Sound, New Zealand [22].

Karate is a small network representing the friendships between 34 members of a karate club at a US university during two years [23]. It has 78 edges.

4.3 Experimental Results

In this part, we present the quality metrics for the community structure in which the percentage $f = [0, 1, 2, 5, 10, 15, 20, 25, 30, 40, 50]$ of edges were replaced. $f = 0$ means that there is no change to the original networks.

Figure 1 shows the results for the Gowalla dataset. The horizontal lines show the quality metric of the community structure detected in the original unchanged network. All three link predictors improved the networks according to eight community quality metrics. PropFlow performs extremely well on Gowalla, except for the Expansion measure. The improvement goes beyond values of $f \leq 50$ reported here. We can observe a limit (varying for different link prediction metrics) of how many links could be replaced for the purpose of improving the community structure of the

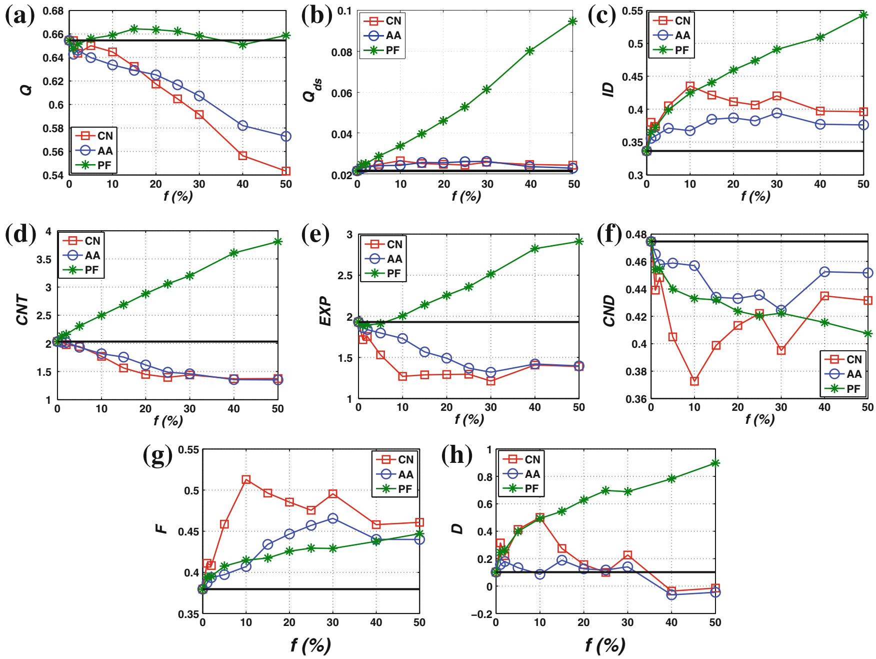

Fig. 1 The quality metrics for the community structure that SpeakEasy discovers on the networks generated from Gowalla using our link improvement method. **a** Q. **b** Q_ds. **c** Intra-density. **d** Contraction. **e** Expansion. **f** Conductance. **g** Fitness. **h** Modularity degree

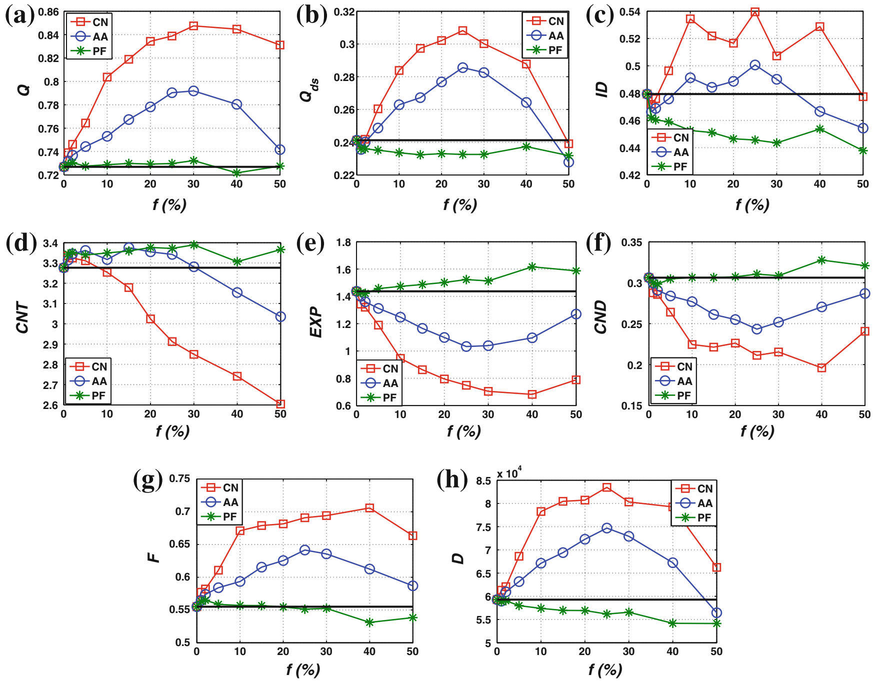

Fig. 2 The quality metrics for the community structure that SpeakEasy discovers on the networks generated from Amazon using our link improvement method. **a** Q. **b** $Q_{d}s$. **c** Intra-density. **d** Contraction. **e** Expansion. **f** Conductance. **g** Fitness. **h** Modularity degree

network. Figure 1b shows that the value of modularity density grows by an order of magnitude compared to the original value with PropFlow.

Figure 2 presents the results of the Amazon dataset. We can observe that CN and Adamic-Adar metrics work well on this network.

Figure 3 displays the results of DBLP. With *CN*, the values of Q_{ds}, *ID*, and *CNT* generally decrease as the replacing percentage f increases. While Q achieves its maximum at $f = 30$, *EXP* and F reach their optima at $f = 40$, and *CND* and D attain theirs at $f = 15$. With Adamic-Adar, the values of Q, *EXP*, *CND*, and F reach their optima at $f = 20$, while Q_{ds} and D achieve theirs at $f = 15$.

Results for smaller networks are summarized in Table 1. This excludes Santa Fe and Karate datasets which, although small, have a well-evolved edge structure and no need for improvement.

Table 1 presents link improvement results for CN and Adamic-Adar metrics on the Amazon, DBLP, Football, and Dolphin datasets. The results for PropFlow are omitted because it works well only on the Gowalla dataset. The cells in the table that contain the fraction of replaced links f show the best f for the corresponding community quality metric. *RI* stands for the relative improvement

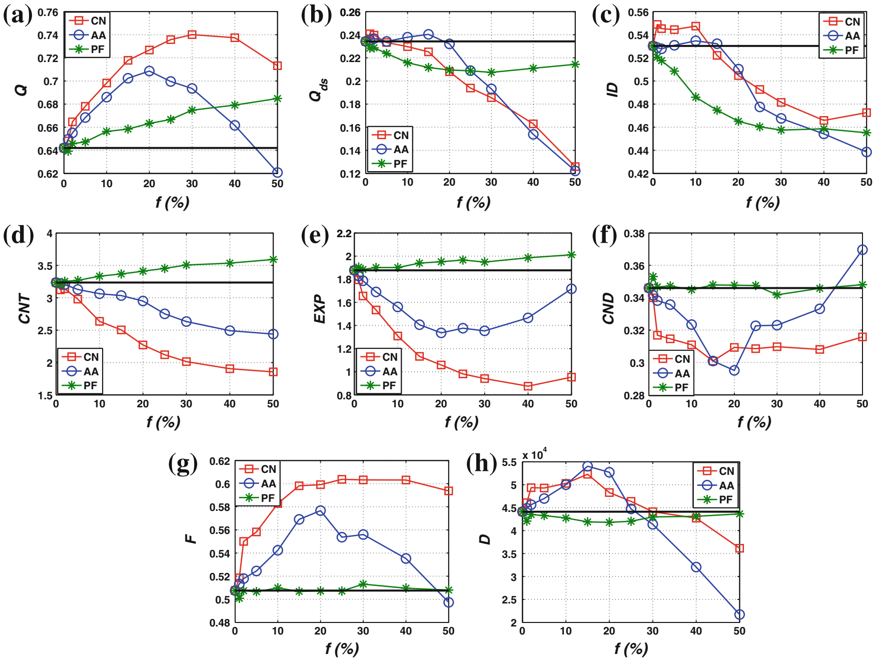

Fig. 3 The quality metrics for the community structure that SpeakEasy discovers on the networks generated from DBLP using our link improvement method. **a** Q. **b** Q_{ds}. **c** Intra-density. **d** Contraction. **e** Expansion. **f** Conductance. **g** Fitness. **h** Modularity degree

of the corresponding community quality metric. It is the percentage of improvement of the metric attained with the best f compared to its original value for the unchanged network ($f = 0$). The ✗ in the table indicates that our link prediction method results in no improvement ($RI < 0$) in that row. This table and all the above figures show that our link improvement procedure is able to significantly refine the community structure of five out of seven networks.

Generally, our method performed best when the number of common neighbors were used as the link prediction metric, followed by the Adamic-Adar metric. Yet for the Gowalla network, the best performing link prediction metric was PropFlow. Therefore, we conclude that a single link prediction metric cannot perform well on all networks. The basic reason for this is that each metric performance depends on the meaning of the relationships which define links in the network. Another reason is that networks also differ in their size, structure, and dynamics. The impact of these factors on link prediction is often unclear. Thus, our method can be used to evaluate the performance of link prediction metrics. If the highly ranked predicted links do not improve quality of communities, they are unlikely to be formed quickly. We also observe that there is a threshold (which varies for different link prediction metrics)

Table 1 The best replacing percentage *f* and the corresponding relative improvement (*RI*) of the community quality metric achieved using the two link improvement method: the number of common neighbors and Adamic-Adar on Amazon, DBLP, Football, and Dolphin

		Datasets							
		Amazon		DBLP		Football		Dolphin	
		f	*RI*	*f*	*RI*	*f*	*RI*	*f*	*RI*
CN	*Q*	30	16.6	30	15.3	***15***	20.5	20	11.7
	Q_{ds}	25	27.8	✗	✗	***15***	34.1	10	25.7
	ID	25	12.6	✗	✗	10	9.8	10	9.4
	CNT	✗	✗	✗	✗	***15***	20.4	25	3.9
	EXP	***40***	***52.6***	***40***	***53.4***	***15***	43.6	***30***	52.3
	CND	***40***	36.0	15	13.0	***15***	41.9	25	42.1
	F	***40***	27.2	***40***	18.8	***15***	35.0	***30***	40.5
	D	25	40.8	15	18.5	***15***	***69.2***	25	***73.0***
AA	*Q*	***30***	8.9	***20***	10.4	15	16.9	15	7.1
	Q_{ds}	25	18.4	15	2.6	15	33.2	10	20.8
	ID	25	4.4	✗	✗	15	9.1	✗	✗
	CNT	15	3.0	✗	✗	***20***	19.4	✗	✗
	EXP	25	***28.2***	***20***	***28.8***	***20***	36.0	20	33.4
	CND	25	20.5	***20***	14.7	***20***	36.5	***30***	20.2
	F	25	15.6	***20***	13.6	***20***	29.1	***30***	18.8
	D	25	26.0	15	22.5	15	***60.1***	15	***40.4***

of how many links could be replaced for the purpose of improving community structure of a network. Going beyond this threshold may lead to higher cost and lower performance although the quality of the community structure may still be better than that of the original unchanged network.

4.4 Impact of the Community Detection Algorithm

To test how much our outcomes depend on the choice of the community detection algorithm, we present here the results of the experiments on one of the largest datasets, Amazon, using GANXiS algorithm [14]. Figure 4 shows the experiment outcomes that are qualitatively similar to those reported in Sect. 4. The scale of improvements and the range of percentages *f* over which the improvements are seen are very similar, while the order of the link prediction methods sorted by their performance remains the same. This is a clear indication that switching to a different community detection algorithm did not impact our conclusions about the use of link prediction methods on the Amazon dataset.

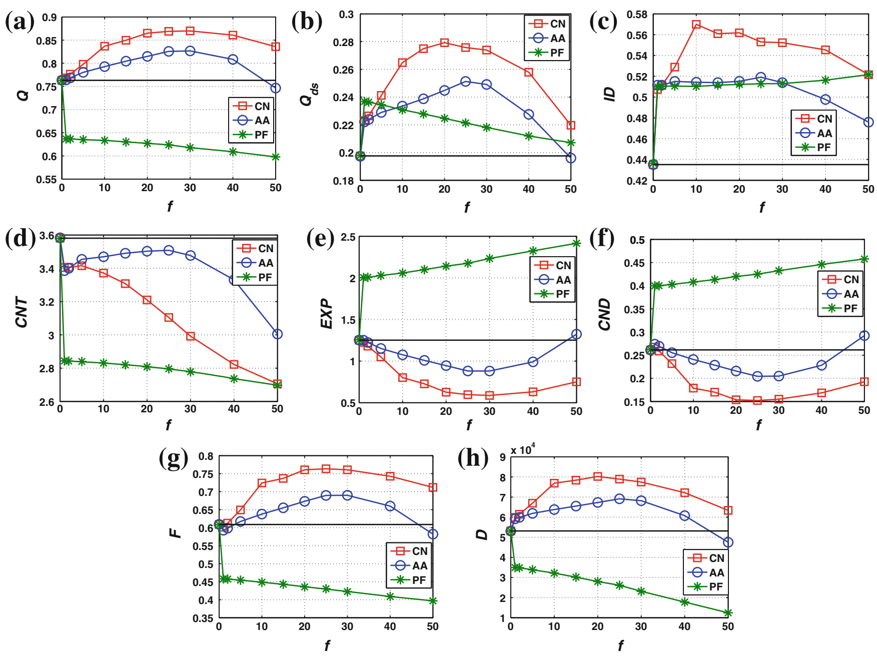

Fig. 4 The quality metrics for the community structure that GANXiS discovers on the networks generated from Amazon using our link improvement method. **a** Q. **b** $Q_{d}s$. **c** Intra-density. **d** Contraction. **e** Expansion. **f** Conductance. **g** Fitness. **h** Modularity degree

5 Evolving Networks

In search for ground truth examples to further verify our methodology, we apply it to social networks evolving from their initial state. This evolution happens when a group of people is put together and makes initial links which are being refined later based on increasing members knowledge about each other. Examples are when middle school students move to high school or later when they join universities. Links from the initial period that are a miss dissolve and are replaced by other links while some initial links persists. The final stage is a stable community, like researchers in Santa Fe or Karate club members. This is distinct from truly dynamic networks where change is constant. So here we are not trying to detect changing points in evolving networks or evolving community structure in dynamic networks. The networks that we consider evolve from an initially suboptimal state to a final stable state. By observing evolving networks at different stages, we are able to measure how our predictions of edge creation and persistence based on the past network states compare to the future state. We use the ArnetMiner paper citation and author collaboration dataset [24]. We restrict the network to a subgraph whose nodes represent authors only in the United States, since collaboration requires language and location affinity.

We divide the dataset into three subsets defined by time periods containing collaborations between 1995–2004, 2005–2009, and 2010–2014. To compare collaborations within any two time periods, we first find the subgraphs for each period that contain an intersection of nodes of both graphs. For the chosen periods, the subgraphs have N = 18,382 nodes and E_o = 44,182, E_n = 43,267 edges in the older and newer subgraph, respectively. The newer subgraph has EN = 28,996 existing new edges and EP = 14,271 existing edges persisting from the older subgraph. With these values, the fraction of new edges in a random sample of edges non-existing in the older graph is $\frac{2*EN}{N(N-1)-2*E_o}$ which is 0.017 % for our graph.

Next, non-existing but highly ranked edges are added as new edges to the older subgraph while low ranking existing edges are removed from it. Then using the newer graph as ground truth, we count how many new edges in the older subgraph actually exist in the newer one. Creating new edges successfully requires a metric consistent with the meaning of the links, which in our case is co-authorship. Hence, in addition to the CN metric, we also introduce a new metric, *Complete Recent Triangles* (CRT) for ranking edges. CRT first identifies all new triangles that are created by adding a new edge to the network. For each such newly created triangle, the CRT metric increases the score of the new added edge by the sum of weights of the two previously existing edges of the triangle. The weight of each such edge is the sum of the recency values of papers co-authored by the authors represented by the edge endpoints. Four age categories are set; less than 2, 2–4, 4–6 years, and older than 6 years. The corresponding values of recency are 1, 0.8, 0.65, and 0.5.

To measure the accuracy and coverage of edges selected as new or persistent by our method, we vary f from 1 to 50. The results are computed with the older period set to 1995–2004, and the newer one set to 2005–2009. The results based on other periods are qualitatively similar. Then, for each f, we compute the numbers of all edges selected as new by the link prediction SN(Selected and New), and the number of such edges that actually exist in the newer subgraph is denoted SEN. The ratio of SEN to SN measures the quality of selection of new edges while the ratio of SEN to EN tells us what percentage of the new edges is covered by the selected new edges. Table 2 shows the results. For the ArnetMiner networks generated for the periods selected for the reported experiments and for CRT, the first ratio varies between 12.22–17.91 %, thus it is up to 1,000 times greater than the fraction of new edges in a random sample of edges. The second ratio shows that the coverage of new edges reaches up to 9.3 % for $f = 50$. The results for CN are worse, the first ratio peaks at $f = 5$ and yields 13.8 % while the coverage peaks at 8.4 % for $f = 50$. For the middle range of f the two metrics perform similarly.

For ArnetMiner network, we used edge persistence selection which is complementary to edge deletion considered for the other network. Like previously, we first rank all existing edges using link prediction method, here CN and CRT. Then we remove $100 - f$ percentage of the lowest ranking edges, thus preserving f percentage of existing edges as persistent. SP denotes the number of existing edges selected as persistent, while SEP is the number of those edges that actually exist in the newer subgraph.

Table 2 Results of predicting added and deleted edges in evolving networks

Measurements	f									
	1	2	5	10	15	20	25	30	40	50
SEN/SN for CN	9.30	12.91	13.81	13.67	13.47	13.07	12.13	11.54	10.68	11.08
SEN/SN for CRT	18.14	17.10	16.03	14.24	13.29	12.78	12.97	12.86	13.02	13.42
SEN/EN for CN	0.14	0.39	1.05	2.08	3.08	3.98	4.62	5.28	6.51	8.44
SEN/EN for CRT	0.28	0.52	1.22	2.17	3.04	3.89	4.94	5.88	7.94	10.23
SEP/SP for CRT	79.86	72.51	64.16	58.68	54.01	51.92	49.32	47.11	44.27	42.71
SEP/EP for CRT	1.18	2.16	4.87	9.23	13.33	17.88	22.41	27.27	39.00	54.68

All ratios are represented as percentages

The results with CN for edge persistence are at the level of random chance, and they are clearly impacted by the massive number of deleted edges. Therefore, we omitted those results here. However, using CRT we again observe the improvements. When persistent edges are selected randomly, the success rate is 32.3 %. At the same time, using CRT with the two smallest values of f, yields the success rate of over 70 %. The best success rate of 79.9 % is achieved with $f = 1$ which is 2.5 times greater than at random. The coverage of persistent edges reaches 54.7 % for $f = 50$.

6 Conclusion and Future Work

In this paper, we introduce an approach for improving the network community structure by removing a certain fraction of low ranking existing links and replacing them with highly ranked new links. The proposed method significantly improves the community structure of the networks we considered. However, there is a threshold of how many links can be replaced in order to refine the community structure of a network. Going beyond this threshold may lead to higher cost and lower performance.

Generally, the link improvement method using the number of common neighbors for link prediction has the best performance, followed by Adamic-Adar, while PropFlow performs extremely well only on Gowalla dataset. We conclude that a single link prediction method cannot perform uniformly well on every network. Some metrics are more suitable than others for a particular network depending on the nature of the links. This was confirmed by our study of the evolving network in which a new link prediction metric for co-authorship, Complete Recent Triangles, delivered the improvement of three orders of magnitude over randomly selecting new edges. Finally, we observe that there is a correlation between the performance of link prediction improvement and certain network properties. Two influential factors are the network size and the degree to which nodes possess global knowledge about the network structure. To confirm that our conclusions do not depend on the use of a specific community detection method, we processed the Amazon dataset with two community detection algorithms, obtaining similar results.

In the future, we plan to design and adopt more link prediction metrics for our approach to explore their performance on different types of networks. We also plan to explore how much our link improvement method could refine the quality of overlapping community structure.

References

1. Fortunato, S.: Community detection in graphs. Phys. Rep. **486**, 75–174 (2010)
2. Yan, B., Gregory, S.: Finding missing edges and communities in incomplete networks. J. Phys. A **44**, 495,102 (2011)
3. Chen, M., Kuzmin, K., Szymanski, B.: Community detection via maximization of modularity and its variants. IEEE Trans. Comput. Soc. Syst. **1**(1), 46–65 (2014)

4. Yan, B., Gregory, S.: Finding missing edges in networks based on their community structure. Phys. Rev. E **85**(5), 056,112 (2012)
5. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Phys. Rev. E 78, 046,110 (2008)
6. Liu, Z., He, J.L., Kapoor, K., Srivastava, J.: Correlations between community structure and link formation in complex networks. PLoS ONE **8**, 72,908 (2013)
7. Liu, Z., Dong, W., Fu, Y.: Local degree blocking model for missing link prediction in complex networks (2014). arXiv:1406.2203
8. Yan, B., Gregory, S.: Detecting community structure in networks using edge prediction methods. J. Stat. Mech: Theory Exp. **2012**(09), P09,008 (2012)
9. Burgess, M., Adar, E., Cafarella, M.: Link-prediction enhanced consensus clustering for complex networks (2015). arXiv:1506.01461
10. Lichtenwalter, R.N., Chawla, N.V.: LPmade: link prediction made easy. J. Mach. Learn. Res. **12**, 2489–2492 (2011)
11. Adamic, L., Adar, E.: Friends and neighbors on the Web. Soc. Netw. **25**(3), 211–230 (2003)
12. Lichtenwalter, R.N., Lussier, J.T., Chawla, N.V.: New perspectives and methods in link prediction. In: Proceedings of the 16th ACM SIGKDD International Conference Knowledge Discovery and Data Mining, pp. 243–252. New York, NY, USA (2010)
13. Gaiteri, C., Chen, M., Szymanski, B.K., Kuzmin, K., Xie, J., Lee, C., Blanche, T., Neto, E.C., Huang, S.C., Grabowski, T., Madhyastha, T., Komashko, V.: Identifying robust clusters and multi-community nodes by combining top-down and bottom-up approaches to clustering. Scientific Reports 5 (2015)
14. Xie, J., Szymanski, B.K., Liu, X.: SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process. In: Proceedings of the Data Mining Technologies for Computational Collective Intelligence Workshop, pp. 344–349. IEEE (2011)
15. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E **69**, 026,113 (2004)
16. Chen, M., Nguyen, T., Szymanski, B.K.: A new metric for quality of network community structure. ASE Human J. **2**(4), 226–240 (2013)
17. Lancichinetti, A., Fortunato, S., Kertész, J.: Detecting the overlapping and hierarchical community structure in complex networks. New J. Physics **11**(3), 033,015 (2009)
18. Li, Z., Zhang, S., Wang, R.S., Zhang, X.S., Chen, L.: Quantitative function for community detection. Phys. Rev. E **77**, 036,109 (2008)
19. Nguyen, T., Chen, M., Szymanski, B.: Analyzing the proximity and interactions of friends in communities in Gowalla. In: Proceedings of the IEEE 13th International Conference Data Mining Workshops (ICDMW), pp. 1036–1044 (2013)
20. Yang, J., Leskovec, J.: Defining and evaluating network communities based on ground-truth. In: Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics, pp. 3:1–3:8. ACM, New York, NY, USA (2012)
21. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proc. Natl Acad. Sci. USA **99**(12), 7821–7826 (2002)
22. 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)
23. Zachary, W.: An information flow model for conflict and fission in small groups. J. Anthropol. Res. **33**, 452–473 (1977)
24. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: ArnetMiner: Extraction and mining of academic social networks. Phys. Rev. E **78**, 046,110 (2008)