A Study of Explainable Community-Level Features

Sophie Sadler[1], Derek Greene[2], and Daniel Archambault[1]

[1] Swansea University, UK
[2] School of Computer Science, University College Dublin, Ireland

Abstract. Community finding algorithms are complex, stochastic algorithms used to find highly connected groups of individuals in a graph. As with "black-box" machine learning approaches, these algorithms provide little explanation or insight into their outputs. In this novel research paper, inspired by work in explainable artificial intelligence (XAI), we look to develop post-hoc explanations for community finding algorithms. Specifically, we aim to identify features that indicate whether a set of nodes comprises a community or not. We evaluate our model-agnostic methodology, which selects interpretable features from a longlist of candidates, in the context of three well-known community finding algorithms.

Keywords: Network Analysis · Explainability · Community Detection

1 Introduction

Community finding is an important task in network analysis for gaining insight into the network structure. Networks are normally used to represent relational, non-Euclidean data. Data points, known as nodes, are connected by edges which represent relationships between the nodes. Communities are loosely defined as sets of nodes with high connectivity within the community and sparser connections to nodes outside of the community [4]. Identifying communities can provide salient information in many applications, such as public health and computational social science. However, existing community finding algorithms provide little insight beyond the identification of the communities themselves. Since these algorithms are often stochastic, producing potentially different community sets across consecutive runs, and also rarely provide reasoning for these outputs, it remains uncertain as to why the algorithm has identified a certain set of communities. So far, there has been little work to provide further reasoning behind community finding algorithm outputs, which might help an end user to understand the results in a practical context. In particular, community finding algorithms are often used by domain experts with some experience with network analysis, so we try to consider explanations that can assist someone who already has some understanding of complex networks.

In the wider field of machine learning, significant research has been devoted to improving our understanding of model outputs [19]. Since many machine learning algorithms act as a "black-box", providing little explanation after training on sometimes millions of parameters, explainability has become imperative to avoiding hidden biases or incorrect assumptions. Although one approach is to develop inherently "transparent" models (i.e., models where the inner workings are easily understood), "post-hoc" explanations have also been developed as an alternative where this is not possible [7, 15]. This involves generating explanations for outputs after the model has already been trained and applied on the data. One such method is to identify interpretable features which a domain expert can easily recognise and understand.

In this paper, we propose a model-agnostic methodology for identifying interpretable features which are able to distinguish "real communities" and "fake communities". We apply this methodology to three well-known community finding algorithms to generate a short list of interpretable features that can be used to explain their outputs. We found that the most informative features across all algorithms were the cut ratio and the internal-external metric, both defined in section 3. As the number of inter-community edges increase, relative betweenness becomes increasingly important. We envision that this insight could be incorporated into future work on generating visual explanations for the outputs of community finding algorithms.

2 Background and Related Work

LIME [15] (Local Interpretable Model-Agnostic Explanations) is a well known method for post-hoc explanations. The model generates these explanations by perturbing the input data and observing the changes to the output. The relative importance of features is included in the explanation, showing which features contributed to the output. In our work, we adopt a similar use of feature importance to identify the interpretable features which will be of most use in explaining communities. SP-LIME (Submodular Pick LIME) aims to explain the model as a whole, rather than individual outputs. The use of interpretable features in a simpler surrogate model has also been explored by Keane et al. [7], where a complicated neural network is mapped to a simpler case-based reasoning model.

To the best of our knowledge, there has been little work adapting these post-hoc explanation approaches to network analysis, and in particular, community finding. Some work in explainability of unsupervised machine learning has aimed to provide insight into clustering algorithms [13, 12], but not networks or communities. Lanchicinetti et al. [10] compared the performance of community finding algorithms by generating networks using their LFR benchmark [8] with embedded "ground-truth" to identify the best performing algorithms. This dataset generation algorithm produces synthetic networks designed to mimic the structure of real-world networks. One hyperparameter, the mixing parameter μ, is used to determine how well-separated the communities are, as this can vary in real-world networks. At low values of μ, the communities are well-separated, with few edges connecting nodes in different communities. At high values of μ, the communities become harder to identify for the algorithms, as there are more edges connecting nodes in different communities. In our work, we repeat our experiments on networks with different values of μ to see whether the separation of communities affects which features are best able to distinguish between communities and sets of nodes which do not comprise a community.

Further work by Lee and Archambault [11] compared the performance of well-known community finding algorithms to communities labelled by humans. Their findings were in line with those of Lanchicinetti et al.; the same community finding algorithms performed best on both the human-labelled communities and the LFR benchmark communities. Among these were Infomap [16] and the Louvain algorithm [1], which we include in our analysis.

While little work has aimed to explain the communities generated by these algorithms, there has been some exploration into using consecutive runs of stochastic algorithms to generate a more "definitive" consensus clustering for a given network [9]. Other work has explored the consistency of algorithms across several runs [3, 5]. In this work, we aim at providing interpretable features to explain community structure, in a manner that is independent of the choice of algorithm.

3 Methodology

In order to identify the interpretable features which can distinguish between real communities and other sets of nodes, we propose the following methodology. We perform a set of experiments on different algorithms at different network μ values (defined in section 2). For each μ value, a large set of synthetic graphs on which to perform the experiments are generated using the LFR benchmark method. Then for each experiment, we run the chosen algorithm on the set of graphs at the correct μ value. For each graph, we perform 1000 runs of the algorithm, and obtain the set of unique communities found across these 1000 runs. One node may appear in many different communities within this set, as the community structure may have been identified differently across different runs of the algorithm. However, each community will appear only once in the dataset. We then calculate our longlist of features for these communities. This gives us our set of features for the "real community" labelled data points.

Then, we use a rewiring process to adjust the network structure. The features are recalculated for each community on the rewired graph, giving us our set of features for the "fake community" labelled data points. Although the set of nodes in the community remains the same, the structure of the community has changed in the rewiring, resulting in new values for the features. However, due to the one-to-one mapping between the "real community" dataset and the "fake community" dataset, we have balanced classes for a classification problem.

We therefore train a random forest classifier to distinguish between the "real" and "fake" community classes using the features we have precalculated. Finally, we extract permutation importances for each of the input features and perform a statistical analysis to identify which are the most informative.

3.1 Graph Generation

Power analysis determined the number of graphs needed at each of the three μ values (0.2, 0.3 and 0.4) to be 119, which we round up to 120, giving 360 in total. These are generated using the LFR generator in NetworkX [6], using the following hyperparameters: numbers of nodes 1000; τ_1 and τ_2 3 and 2; average degree 20; max degree 50. These were chosen to match those used in the original LFR benchmark paper [10].

3.2 Community Detection

The following stochastic algorithms, designed for detecting partitions (i.e., non-overlapping communities), are used for our experiments: Louvain [1], Infomap [16], Label Propagation (LPA) [14]. In total, this gives 9 potential experiments (3 algorithms on 3 μ values). However, we omit LPA on $\mu = 0.4$ since it often classifies all nodes in the graph as belonging to a single community.

3.3 Calculation of Features

For these experiments, we consider features which are interpretable to domain experts who have previous experience with network analysis. Thus, these features may not be interpretable to a layperson, but should be considerably easier for the expert to understand than the algorithm on its own. We try to consider the simplest features possible, omitting, for example, modularity in favour of those which are easier to understand. The longlist of features chosen for our experiments are as follows:

- *Relative Density*: For a set of nodes V connected by a set of edges E, the density of this set is defined as: $2|E|/(|V|(|V|-1))$. Then the relative density is the density of the community divided by the density of the whole graph.
- *Relative Diameter*: For a set of nodes V, the diameter is the maximum distance between any two nodes. Then the relative diameter is the diameter of the community divided by the diameter of the whole graph.
- *Relative Pathlength*: For a set of nodes V, the average shortest path length is defined as: $\sum_{s,t\in V} d(s,t)/(|V|(|V|-1))$ where $d(s,t)$ is the length of the shortest path from s to t. Then the relative path length is the average shortest path length of the community divided by the average shortest path length of the whole graph.
- *Relative Degree*: For a set of nodes V, the average degree centrality is defined as: $\sum_{i\in V} deg(i)/(|V|(|V|-1))$ where $deg(i)$ is the number of edges adjacent to node i. Then the relative degree is the average degree centrality of the community divided by the average degree centrality of the whole graph.
- *Relative Betweenness*: Let V be a set of nodes of which i, j and k are members. Let $\sigma(j,k)$ be the number of shortest (j,k) paths, and $\sigma(j,k|i)$ be the number of those paths that pass through i. Then the betweenness centrality of node i is given by: $\sum_{j,k\in V} \sigma(j,k|i)/\sigma(j,k)$ Note that if $j = k$, $\sigma(j,k) = 1$

and if either j or $k = i$, then $\sigma(j, k|i) = 0$. Then the relative betweenness is the mean betweenness centrality of the nodes in the community divided by the mean betweenness centrality of the nodes in the whole graph.
- *Relative Closeness*: Let V be a set of nodes of which i and j are members, and let $d(i, j)$ be the length of the shortest path between nodes i and j. Then the closeness centrality of node i is given by: $(|V| - 1)/\sum_{j \neq i} d(j, i)$ Then the relative closeness is the mean closeness centrality of the nodes in the community divided by the mean closeness centrality of nodes in the whole graph.
- *Cut Ratio*: Let $w_{ij} = 1$ if nodes i and j share an edge, and $w_{ij} = 0$ if they do not. Then we calculate a cut ratio where set A is the set of nodes in the community, and set B is the set of nodes in the graph *not* in the community, as follows: $1/|A||B| \sum_{i \in A, j \in B} w_{ij}$
- *Internal-External*: Let $w_{ij} = 1$ if nodes i and j share an edge, and $w_{ij} = 0$ if they do not. Then we calculate a measure of internal-external where set A is the set of nodes in the community, set E is the set of edges in the community, and set B is the set of nodes in the graph *not* in the community, as follows: $|E|/(|E| + \sum_{i \in A, j \in B} w_{ij})$

3.4 Graph Rewiring

In order to rewire the graph to calculate the "fake community" features, pairs of edges are identified randomly and endpoints are swapped. For example, for edges (i, j) and (a, b), after a swap the edges will be (i, a) and (j, b) with the original edges removed. A noise level is multiplied by the total number of nodes in the graph to determine the number of edge swaps. For our experiments, the noise level is 0.5. A swapping probability of 0.5 represents a balance between a uniform structure and a random structure in the original discussion of rewiring by Watts and Strogatz [20].

3.5 Random Forest Classifier

When applying a random forest classifier to distinguish between "real" and "fake" communities, we first split the data into 80% training data and 20% test data. We then train the random forest 50 times on the training data, comprised of 10 repeats of 5-fold cross-validation. A permutation importance is calculated for each node feature after each of these 50 runs, using the held-out test data. These 50 values then give a distribution for the importance of that feature.

3.6 Statistical Methodology

Pilot study. In order to determine the suitability of the above methodology, we ran a pilot study on 20 LFR generated networks at each μ value (60 networks total). This pilot study was used to determine the behaviour of the underlying phenomena so that we could conduct a proper power analysis. Pilot study networks are not included in the final analysis.

Distributions of each permutation importance for all metrics were created and we ran a Shapiro-Wilk test to determine the normality of the phenomena. As the majority of the distributions in the pilot study followed a normal distribution (77%), we took the assumption that the underlying phenomena were normal for our power analysis.

Statistical methodology. Tests with Bonferroni-Holm corrections are used in our analysis. The normality of the final permutation importance values on the main experiment are confirmed using a repeat of the Shapiro-Wilk tests, and thus t-tests are used to compare the features. Power analysis was conducted with the following parameters: Cohen's effect size of 0.3, significance level of 0.05, and a power of 0.9. We treat each of the three community finding algorithms independently for analysis and use the pairwise t-tests to identify significantly different pairs of features.

4 Experiments

In this experiment, we aim to identify which of the features have a significantly greater importance than the others in predicting whether a set of nodes is a "real" or "fake" community. Distributions of permutation importance for the features on each experiment are displayed below in figure 1. These distributions are constructed using the 50 permutation importance values calculated during training, as described in section 3.5.

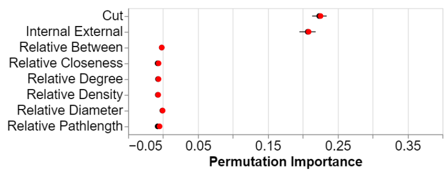
From our experimental results, we see qualitatively that the cut ratio and internal-external metrics are consistently the most important features in distinguishing the "real communities" from the "fake communities". However, as the μ value increases, there is evidence to suggest that the relative betweenness may also have some importance. The Bonferroni-Holm corrected t-tests identified significant differences between these three "important" metrics and all other metrics in the study with the following exceptions:

- Infomap $\mu = 0.2$: relative betweenness compared with relative diameter
- Louvain $\mu = 0.2$: relative betweenness compared with relative diameter
- LPA $\mu = 0.2$: relative betweenness compared with relative degree and relative density

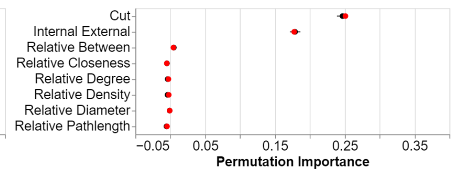
The full set of statistical results are available in the supplementary material.

5 Discussion

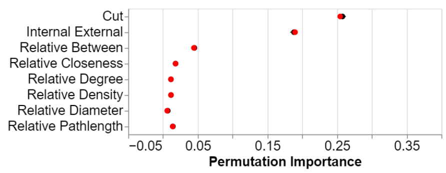
Infomap, Louvain, and LPA performed similarly in our experiments, with three metrics being consistently important for distinguishing the real communities from the fake communities. These metrics were: cut ratio and internal-external for all experiments, with relative betweenness becoming increasingly important



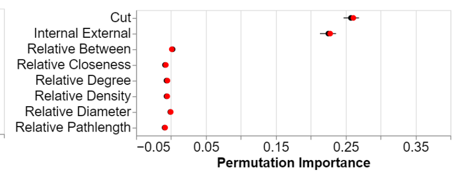
(a) Infomap, $\mu = 0.2$



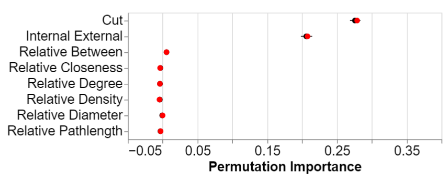
(b) Infomap, $\mu = 0.3$



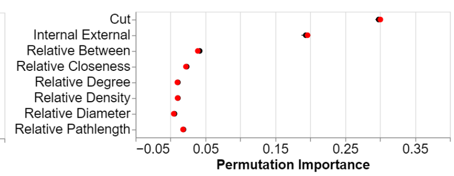
(c) Infomap, $\mu = 0.4$



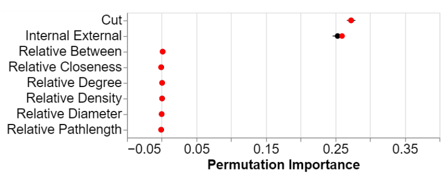
(d) Louvain, $\mu = 0.2$



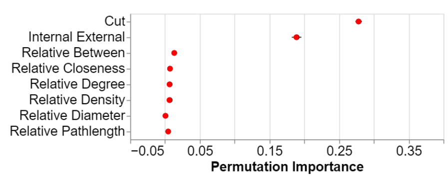
(e) Louvain, $\mu = 0.3$



(f) Louvain, $\mu = 0.4$



(g) LPA, $\mu = 0.2$



(h) LPA, $\mu = 0.3$

Fig. 1: Results of the community feature experiments. Plots are of permutation importance of the metrics. Mean indicated as a black dot and median as a red dot. Lines indicate 95% bootstrapped confidence intervals.

with increasing mixing parameter. Thus, when the communities are more clearly defined, or when μ has a low value, more agglomerative features are important for explaining community structure for the set of nodes considered. This finding is not all that surprising as these metrics are fundamental to the definition of community structure. However, as μ increases and the communities become less well defined, divisive features, such as relative betweenness, become of increasing importance. A possible explanation for this finding is that local connections and neighbours to the node set can be used to understand the community structure when there are few cross community edges, but as these edges increase in prevalence, more global features, such as relative betweenness, become important in explaining community structure.

Surprisingly, despite the fact that the community detection algorithms under consideration involved different objective functions and optimisation strategies, the features that best explained the resulting community structures were the same. This provides evidence that explainable network analysis is feasible, independent of the choice of algorithm. In future work, it would be interesting to construct such systems to explain found community structures to end users.

6 Conclusion

In this paper, we have presented the methodology and results of an experiment to determine features that can be used to explain detected community structure in networks to a domain expert with some understanding of network analysis. Our experiment tested features which were calculated on sets of nodes that could form a possible community. We find that cut ratio and the internal-external ratio are the most informative features when identifying whether a group of nodes is a community. As the mixing of the communities increases, relative betweenness increases in importance. This finding was consistent across all community finding algorithms, indicating the potential for using the proposed method to generate model-agnostic explanations. Moving forward, we would like to perform the necessary HCI and visualisation work to construct explainable network analysis systems that help real users with their tasks. In particular, understanding community structure in social networks is a problem of importance for public health researchers studying social contagion [18, 2] and planning interventions [17] for preventing the spread of harmful behaviour.

Acknowledgements. This work is supported by the UKRI AIMLAC CDT, funded by grant EP/S023992/1 and by Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289_P2.

References

1. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment **2008**(10), P10008 (2008)

2. Brown, R.C., Fischer, T., Goldwich, A.D., Keller, F., Young, R., Plener, P.L.: #cutting: Non-suicidal self-injury (NSSI) on Instagram. Psychological Medicine **48**(2), 337–346 (2017)
3. Chakraborty, T., Srinivasan, S., Ganguly, N., Bhowmick, S., Mukherjee, A.: Constant communities in complex networks. Scientific reports **3**(1), 1–9 (2013)
4. Fortunato, S.: Community detection in graphs. Physics reports **486**(3-5), 75–174 (2010)
5. Francisco, A.P., Oliveira, A.L.: On community detection in very large networks. In: Complex Networks 2011. pp. 208–216. Springer (2011)
6. Hagberg, A., Swart, P., S Chult, D.: Exploring network structure, dynamics, and function using networkx. Tech. rep., Los Alamos National Lab., NM (2008)
7. Keane, M.T., Kenny, E.M.: How case-based reasoning explains neural networks: A theoretical analysis of XAI using post-hoc explanation-by-example from a survey of ANN-CBR twin-systems. In: Proc. International Conference on Case-Based Reasoning (ICCBR'19). pp. 155–171. Springer (2019)
8. Lancichinetti, A., Fortunato, S.: Community detection algorithms: A comparative analysis. Physical Review E - Statistical, Nonlinear, and Soft Matter Physics **80**(5), 1–12 (2009)
9. Lancichinetti, A., Fortunato, S.: Consensus clustering in complex networks. Scientific reports **2**(1), 1–7 (2012)
10. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Physical Review E - Statistical, Nonlinear, and Soft Matter Physics **78**(4), 1–6 (2008)
11. Lee, A., Archambault, D.: Communities Found by Users – not Algorithms. In: Proc. 2016 CHI Conference on Human Factors in Computing Systems. pp. 2396–2400 (2016)
12. Loyola-Gonzalez, O., Gutierrez-Rodríguez, A.E., Medina-Pérez, M.A., Monroy, R., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A., Garcia-Borroto, M.: An explainable artificial intelligence model for clustering numerical databases. IEEE Access **8**, 52370–52384 (2020)
13. Morichetta, A., Casas, P., Mellia, M.: EXPLAIN-IT: towards explainable AI for unsupervised network traffic analysis. In: Proc. 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks. pp. 22–28 (2019)
14. Raghavan, U.N., Albert, R., Kumara, S.: Near linear time algorithm to detect community structures in large-scale networks. Physical review E **76**(3), 036106 (2007)
15. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should I trust you?" Explaining the predictions of any classifier. Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining pp. 1135–1144 (2016)
16. Rosvall, M., Bergstrom, C.T.: Maps of random walks on complex networks reveal community structure. Proc. National Academy of Sciences of the United States of America **105**(4), 1118–1123 (2008)
17. Valente, T.W.: Network interventions. Science **337**(6090), 49–53 (2012)
18. Valente, T.W., Yon, G.G.V.: Diffusion/contagion processes on social networks. Health Education & Behavior **47**(2), 235–248 (2020)
19. Wachter, S., Mittelstadt, B., Russell, C.: Counterfactual explanations without opening the black box: Automated decisions and the GDPR. Harv. JL & Tech. **31**, 841 (2017)
20. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature **393**, 440–442 (1998)