IMA Journal of Complex Networks (2013) Page 1 of 8
doi:10.1093/comnet/xxxxxx

Community detection: effective evaluation on large social networks

Conrad Lee
Clique Research Cluster
University College Dublin
8 Belfield Office Park, Clonskeagh
Dublin 4, Ireland
conradlee@gmail.com

AND

Pádraig Cunningham
Clique Research Cluster
University College Dublin
8 Belfield Office Park, Clonskeagh
Dublin 4, Ireland
padraig.cunningham@ucd.ie

[Received on XX XX XXXX; revised on XX XX XXXX; accepted on XX XX XXXX]

While many recently proposed methods aim to detect network communities in large datasets, such as those generated by social media and telecommunications services, most evaluation (i.e., benchmarking) of this research is based on small, hand-curated datasets. We argue that these two types of networks differ so significantly that by evaluating algorithms solely on the smaller networks, we know little about how well they perform on the larger datasets. Some recent work addresses this problem by introducing social network datasets annotated with meta-data is believed to approximately indicate a "ground truth" set of network communities. While such efforts are a step in the right direction, we find this meta-data problematic for two reasons. First, in practice, the groups contained in such meta-data may only be a subset of a network's communities. Second, while it is often reasonable to assume that the meta-data is related to network communities in some way, we must be cautious about assuming that these groups correspond closely to network communities. Here we consider the difficulties associated with benchmarking community detection algorithms using meta-data that suffers from these two problems, and propose an evaluation scheme based on a classification task that is tailored to deal with them.

Keywords: social networks, community detection, evaluation, benchmarking

1. Introduction

Community structure has been identified as playing a key role in the formation and function of many systems, so it comes as no surprise that there are hundreds of papers currently published on the topic every year. However, in the literature it is commonly observed that although many community detection methods exist, we do not know which ones work best on real data [11, 19, 34, 35]. This problem of evaluation on real data is so acute that, in what has been recognized as the authoritative review on the community detection problem, Fortunato states that our inability to properly benchmark algorithms has led to "a serious limit of the field" and that little is known about which methods perform best in practice [11].

For some types of data, progress has been made by creating benchmarks which suitably incorporate meta-data; for example, in protein-protein interaction networks, the quality of a community can be evaluated by measuring how enriched its members are in terms of its gene ontology annotation. [1, 2, 23]. In this paper, we focus on *social networks* mined from sources such as Facebook or mobile communication logs, and consider the particular challenges that arise when one tries to benchmark community detection algorithms using the meta-data associated with such networks.

In section 2, we motivate the present work by arguing that standard social network benchmarks based on small, purpose-gathered datasets such as Zachary's Karate club tell us little about how a method performs on larger, mined datasets. We also briefly cover recent work on evaluation, which is mostly based on either synthetic (i.e., artificially created) network data, or on non-social networks.

In section 3, we establish some terminology and then consider the specific difficulties that arise when trying to evaluate an algorithm's accuracy using imperfect meta-data. We describe two common deficiencies that meta-data often suffer from: (1) The meta-data is incomplete in that it does not include all valid network communities. Any evaluation scheme based on such an attribute will therefore be limited to measuring recall and cannot measure precision. (2) Groups referred to by the meta-data may be super-sets of network communities—in other words, several valid network communities may be nested with each of the groups in the meta-data. Thus, while the meta-data is closely related to network communities, it is at a coarser granularity.

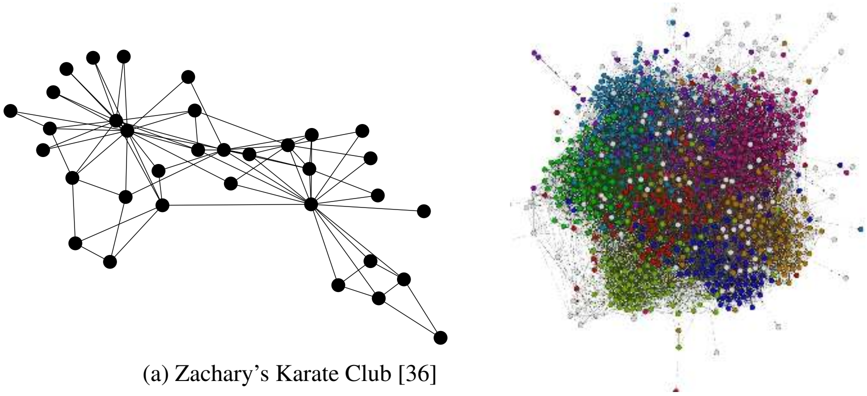
In section 4, we propose a classification-based evaluation scheme which we believe appropriately utilizes such imperfect meta-data for effective benchmarking. In this evaluation scheme, we use the communities detected by an algorithm to infer the value of node attributes related to community structure. The inference is done in a supervised machine-learning setting where the community assignment matrix provides the features associated with each node. The idea is that, if the community detection algorithm has done a good job, then a machine learning classifier should be able to use the community assignment matrix to accurately infer such an attribute.

2. Motivation: Mined data differs from purpose-gathered data

From the 1940s to the 1990s, the datasets used to evaluate community detection algorithms were generally *purpose-gathered* for research by experts who had first-hand knowledge of the social system with which the dataset was associated. Examples include Moreno's early datasets dating back to the 1920s [10, 25], the Southern Women dataset [6], Sampson's Monks [30], and Zachary's Karate Club [36]. Through their close observation, the researchers who collected this data were able to group the nodes into communities based on events such as crises or social gatherings. During this time, network datasets tended to be small (with fewer than 500 nodes, and often fewer than 50 nodes) and well-studied (in [12], Freeman synthesizes the findings of 21 methodological studies on the Southern Women's network alone).

A new era of work on community detection began in the late 1990s, created in part by a new type of *mined* social network data that was extracted, for example, from mobile communication records or Facebook interactions [11]. Figure 1 displays an example of both a purpose-gathered and a mined network. While mined data still represents social networks, we posit that it differs from purpose-gathered data in important ways. Our motivation is not to claim that one type of data is superior to the other; rather, we wish to show that for the purpose of evaluating the effectiveness of community detection methods, it makes sense to distinguish between the two. In particular, we contend that even if an algorithm works well on purpose-gathered datasets like Zachary's Karate club, we may nevertheless have little idea of how well it performs on mined datasets.

Here we summarize three important differences between the two types of data related to (1) unifor-



(a) Zachary's Karate Club [36]

(b) Facebook friendships at Caltech [31]

FIG. 1: On the left, a network which is typical of *hand curated* datasets gathered by a researcher in the field. On the right an example of a *mined* dataset. We show that the structure of the communities in these two types of networks differs in important ways.

mity of coverage, (2) overlapping social contexts, and (3) size:

1. With regard to **uniformity of coverage**, we observe that a purpose-gathered dataset's careful curation tends to ensure that sufficient information is gathered on each participant, or at least on most participants. In mined datasets, on the other hand, it is not uncommon for the majority of users to have very low activity levels and thus, arguably, be left out of the dataset entirely. Many mined datasets contain a large proportion of these low-degree nodes, leading Leskovec et al. to characterize them in terms of whiskers and cores [22].

2. Concerning **overlapping social contexts**, we point out that the purpose-gathered datasets typically cover only one social context, such as activity in a club, at home, or in the workplace. In contrast, mined data from services like Facebook combines interactions from several social contexts, such as personal and professional life, jumbling them together. By combining several contexts into one network, a typical node may belong to several communities, leading to a condition which has been called *pervasive overlap* and which has been shown to cause many community detection algorithms to perform poorly [1, 14, 20, 24, 28].

3. Finally, the **size** of mined datasets is often several orders of magnitude larger the size of purpose-gathered datasets. This is an important difference between the two types of datasets because the performance of many community detection methods, such as those based on modularity maximization, has been shown to decrease as the size of the network increases. [17]

Any one of these differences may cause an algorithm that works well on purpose-gathered datasets to fail on mined datasets. In one sense, researchers recognized that new types of data required new approaches: the emergence of mined datasets inspired many new methods for community detection. Indeed, the field of community detection enjoyed booming popularity as more physicists, computer scientists, and social scientists developed these new methods, perhaps motivated by the potential discoveries that could be made in the mined data. [27, 32]

However, we argue that when modern community detection methods were evaluated on social network data something went wrong: rather than evaluating these new methods on the mined datasets for which they were designed, the new methods were often evaluated on the old, purpose-gathered datasets. [5, 8, 13, 26][1] Thus, we know that many of the new community detection methods work well on datasets like Zachary's Karate Club or the Southern Women's dataset, but we do not know how well they work on larger, digitally extracted datasets, and this is the ignorance that Fortunato described in the excerpt above.

2.1 *Existing work on evaluation*

Our point in the preceding paragraphs is that evaluation on large mined *social* network datasets is lacking. However, much work has gone into evaluating these methods on *other types* of large, mined datasets, such as biological and synthetically created networks, and here we briefly review that work.

Community detection methods have been evaluated on diverse types of data. For example, the Gene Ontology and other annotation can be used to evaluate the modules found in protein-protein interaction networks [23] and product categorizations can be used to annotate the network of products co-purchased on Amazon.com [35]. See [1] for an example of thorough benchmarking of community detection methods on many types of large, mined datasets.

With the introduction of the LFR specification in [15, 18], the last five years has also seen progress in terms of synthetic benchmarking. In a synthetic benchmark, a network is artificially created and a known set of ground truth communities is planted into it; see [14, 16, 21, 33] for comparative evaluations based largely on synthetic benchmarks. The advantage of this approach is that because the ground-truth set of communities is known, the evaluation procedure is clear-cut. The disadvantage of this approach is that synthetic networks may differ from empirical networks in important but unknown ways—if the field depends too heavily on synthetic benchmarks, then it may develop methods that work well on fake data but poorly on real data.

There is thus a clear need for benchmarks based on mined social network data. Two recent papers from Yang and Leskovec acknowledge this need and contribute large social network datasets with meta-data which they claim approximates a ground-truth of network communities [34, 35]. Similarly, a high-quality ensemble of 100 Facebook networks was released by Traud et al. in [31], which includes node attributes that are considered to be closely related to community structure. If we are to make progress in detecting community structure in modern datasets, then it is imperative that we design benchmarks based on such data. However, as we will see in the next section, the specification of such benchmarks is not straightforward because we cannot assume that the meta-data associated with these datasets corresponds to the "true" network communities that an ideal algorithm would find.

3. Imperfections in meta-data: incompleteness and nesting

We have so far motivated this paper with the observation that if we want to measure how well community detection algorithms perform on modern mined networks, then we must evaluate them on similar mined networks rather than on small, hand-curated datasets. We now deal with the problem of carrying out evaluation with mined data for which no perfect ground truth exists.

[1] In some of these papers a larger social network was evaluated (such as the co-authorship network on arXiv), but these lacked the ground-truth or meta-data necessary for a proper evaluation. These larger networks were typically employed only for comparing something other than how well the algorithm identifies all relevant community structure, such as which algorithm gets the highest modularity or runs quickest.

The famous opening lines to Tolstoy's *Anna Karenina* state that "Happy families are all alike; every unhappy family is unhappy in its own way." This is also true of the meta-data used to evaluate clustering algorithms, which can be deficient in many different ways. Depending on the exact nature of these deficiencies, one may need to use a completely different evaluation scheme in order to achieve effective evaluation. Here we focus on the case in which we have an attribute which we assume is closely related to community structure, but which suffers from two deficiencies: incompleteness and nesting. Before describing these deficiencies in detail, it will be useful to establish some terminology.

3.1 *Terminology*

We will define the *true communities* as the ideal output of a community detection algorithm.[2] We will use the term *meta-data* as a catch-all for all of the node attributes. Meta-data often has many attributes, where each maps nodes to some class or set of classes. For example, the gender attribute maps each node to a single class (i.e., a value), such as male or female. Other attributes might map each node to a multiple classes; for example, in [34, 35] each node can belong to zero or more *user-defined* groups—in this case, a node could belong to several groups corresponding to hobbies. In general then, each attribute maps a node to a set of classes–we will refer to these as *attribute classes*.

If we run a community detection algorithm on a network, it returns a list of communities which we will call the *found communities.* Note that both an attribute and a set of found communities map each node to zero or more classes, so it is possible to measure the similarity between the two.

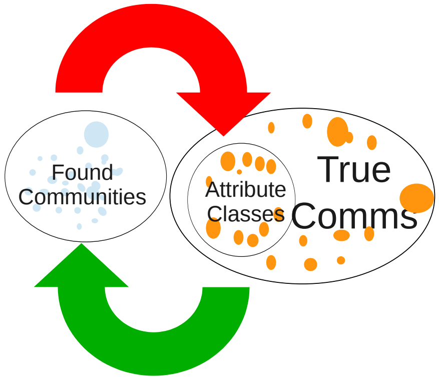
When we describe the quality of a set of found communities, we will do so in terms of *precision* and *recall*. Precision measures the fraction of the found communities that exist in the true communities, whereas recall measures that fraction of true communities that exist in the found communities. For example, imagine a situation where there are 100 true communities, but a community detection algorithm finds only ten of these (and no extraneous communities): in this case, the precision is 100% while the the recall is 10%.

3.2 *Incompleteness and its consequences*

Let us imagine that we have a collegiate social network that we want to use to evaluate the quality of a community detection algorithm. Assume that in addition to the social network, we have data on the dormitory (henceforth, dorm) attribute, which maps each student to the dorm he or she lives in. For now let us assume that each dorm exactly corresponds to a true community (we will relax this assumption below). Let us also assume that we know that in addition to those based around dorms, there are network communities formed by other aspects of social life, such as hobbies and academic pursuits. However, in this imaginary scenario, we have do not have data on these other attributes. The right side of fig. 2 depicts this situation. Thus, the dorm attribute is an *incomplete* enumeration of the true communities.

Now imagine that we run a community detection algorithm on this network and are left with a set of found communities, represented by the blue circles in fig. 2. Given that the dorm attribute is incomplete,

[2] One might object to such an idea, and argue that there is no objectively correct set of true communities in a network—that depending on the purposes for which one wishes to use them, there may be several valid, yet distinct, sets of communities in the same network. This is a reasonable objection, but here we simply assume that there is one set of communities that is most generally useful for a wide range of purposes, and this is what is meant we refer to the true communities. While this assumption may seem dogmatic, it underpins the entire literature on community detection, and it is not an assumption which we will debate here.



Not every found community should be in the set of attribute classes, because the attribute classes do not cover all true communities.

Each attribute class should have a matching found community.

FIG. 2: When we evaluate the accuracy of a community detection algorithm, the attribute data we use may contain only a subset of the true communities. In this case, we cannot demand that each found community corresponds to an attribute class, and so we cannot measure the algorithm's precision. We can however demand that each of the attribute classes correspond to one of the found communities, and so we can measure the algorithm's recall.

one might ask whether it is even possible to evaluate the quality of the found communities. In the terminology defined above, we are able to measure the recall of the community detection algorithm, but not its precision. That is, for each dorm, we know that there should be a matching found community—if there is not, we can say that the algorithm failed to detect a community it should have detected. However, if a found community has no matching dorm, then the situation is ambiguous. It could be that the found community is spurious and does not correspond to a true community, or it may be that while it is a true community, it is not a dorm.

Any evaluation scheme based on an attribute that is incomplete in this way, including the scheme we propose below, will have the following limitation: while it it may be able measure an algorithm's recall, it cannot measure its precision. Given that this problem has plagued the related field of clustering for decades [9], researchers wishing to evaluate community detection algorithms with mined data may have to simply accept this limitation.

3.3 *Network communities may be nested in attribute classes*

The example above included a collegiate social network well as knowledge of each node's dorm, and for simplicity's sake we assumed that each dorm corresponded exactly to a network community. In practice, however, this may be bad assumption. Even if we have an attribute such as dorm that we know to be closely related to the formation of network communities, it may be that there is not a one-to-one correspondence between the two. In particular, network communities may be *nested* within the attribute's classes. For example, it could be that within each dorm, each incoming freshman class forms its own network community. In this case, the network communities would correspond to the intersection

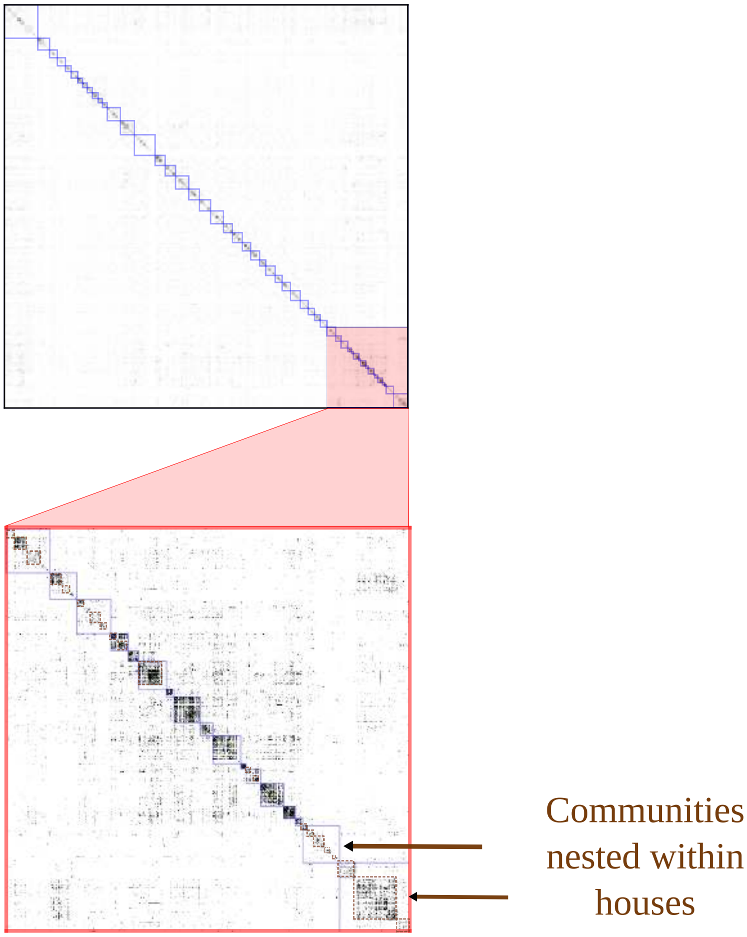


FIG. 3: Upper pane: the adjacency matrix of the University of Chicago, with house membership highlighted (zoomable online). Lower pane: a zoomed in region suggests that sub-communities exist within houses, and that macro-structure exists between houses.

of the dorm and year attributes.

Figure 3 indicates that this example of nesting is not only a hypothetical problem, but one which exists in real data, in this case the University of Chicago's Facebook friendship network from the Facebook100 dataset [31].[3] On the face of it, it seems reasonable to set up a benchmark in which the goal of the community finding algorithm is to detect the "houses" (residential housing units), which are

[3] To create this figure, we have first taken the Facebook friendship network of the University of Chicago and arranged the adjacency matrix such that all of the nodes that belong to the same dormitory are placed in contiguous blocks, indicated by the blue squares along the diagonal. Within each block, the nodes are arranged according to the communities found on the sub-graph induced by the nodes in the block. Furthermore, the blue blocks themselves are ordered according to the communities found in the "meta-graph" formed by collapsing each of the blue blocks into a single meta-node and aggregating links between the meta-nodes. For both the sub-graph and the meta-graph, the Louvain method of community detection was used [4].

displayed along the diagonal in blue. According to the University of Chicago website, “each house represents a tight-knit community of students, resident faculty masters, and residential staff, who live, relax, study, dine together at House Tables, engage, socialize, and learn from each other.”

However, the bottom panel of fig. 3 suggests that while houses are indeed closely related to community structure, they *do not* correspond to network communities. In fact, there may be several times more network communities than houses, and that in general each house is several times larger than a network community. In most cases a single house (indicated in blue) appears to contain many separate, densely connected sub-graphs (in brown) which seem to be network communities. While we have highlighted only a few of these sub-groups, if one zooms into the top panel of fig. 3 and carefully examines the houses highlighted in blue along the diagonal, one can observe that in general each house seems to contain many cohesive sub-graphs.

4. A classification-based evaluation to cope with imperfect meta-data

We have just described two deficiencies which mined datasets may suffer from: incompleteness and nesting. We now propose that by utilizing an evaluation framework which incorporates a machine learning classifier, one can measure the recall of a community detection algorithm. The basic idea behind this evaluation scheme is that, with a bit of simple logic, one can map network communities to attribute classes, even if the former are nested within the latter. Given a good set of communities a machine learning classifier should be able to discover this mapping.

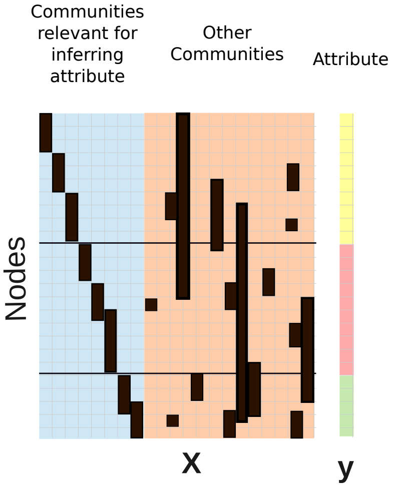


FIG. 4: A machine classifier can detect which network communities are relevant for inferring an attribute.

An example is illustrated in fig. 4. We can represent the mapping from node to found community using the community assignment matrix **X** (sometimes called the indicator matrix), and we can represent the attribute that we use for evaluation as a vector **y**. The task of the classifier is to infer **y** from **X**. Because the attribute we use for evaluation is incomplete, many of the true communities are not relevant for inferring the attribute value—hence, a good classifier should learn to ignore the communities represented by red columns in fig. 4. The classifier does not assume a one-to-one correspondence between the attribute classes and the communities, but rather can flexibly learn the correct mapping between them. In this case, the classifier should learn that the communities corresponding to the first three columns of **X** map to the same attribute value.

Thus, given a complete set of the true communities, the classifier will be able to learn how to infer **y** from **X**. On the other hand, a community detection algorithm with poor recall will create an **X** which does not contain the information necessary to infer **y**. For example, if many of the blue columns were missing from the blue section of fig. 4, then the classifier would become less accurate.

In this context, the classifier is a black box which should perform two functions: ignore features (i.e., communities) in **X** which are irrelevant for inferring the attribute **y**, and flexibly learn how membership in various communities is correlated with **y**. Machine learning classifiers are designed to deal with the former problem by performing feature selection, and the latter by having an expressive hypothesis space. Thus, the classifier should both perform well at feature selection and have a hypothesis space expressive enough to account for the fact that some network communities may be subsets or super-sets of the classes referred to by the attribute. There are many classifiers that fulfill these requirements, including ensemble techniques based on decision trees, such as random forests or stochastic gradient boosting. For those unfamiliar with supervised machine learning, we recommend [3].

There are some limitations and downsides to this classification-based evaluation. First and foremost, it can only measure recall, and not precision, as mentioned above. Additionally, it is only appropriate for the case where the case where we can assume that network communities are cleanly nested within attribute classes. While that appears to be the case for both the dormitory and year attributes in the previously mentioned Facebook100 networks, it is difficult and rather subjective to judge whether a given dataset fulfills this requirement.

Another downside is that setting up such classification tasks is rather complicated. One must ensure that the classifier performs well both at feature selection and at learning the relationships between network communities and the attribute used for evaluation. If the classifier does not perform well at these two tasks, then we can conclude little from the evaluation: if the resulting classification accuracy is low, then one will not know whether it was because the network communities are uninformative with respect to the attribute (and thus a bad set of communities), or whether the classifier was to blame. Further complexity creeps in when one evaluates the performance of a classifier, which typically involves *k*-fold cross validation. In practice, yet another complication is added by computational complexity: depending on the classifier one uses, the size of the network, and the number of communities found, it can be computationally expensive to train a classifier.

5. Conclusion

In section 2, we distinguished between mined networks and small, hand-curated networks, and we argued that these two types of data differ in important ways. We also pointed out that although recently introduced methods of community detection are supposed to work on digitally extracted networks, in practice the only real datasets they are tested on are small and hand-curated. As a result, we are unaware of how well these methods work on larger, mined networks.

This ignorance is caused in large part by the lack of mined networks with acceptable ground-truth data. In section 3 we outlined two particular imperfections that meta-data may suffer from in an evaluation context: incompleteness and nesting. If a node attribute is incomplete (i.e., the attribute classes do not refer to all network communities), then any evaluation is carried out with it will be limited to measuring recall, and will not be able to measure precision. We also demonstrated the problem of nesting with an empirical example: we showed that network communities do not appear to correspond to the the residential houses in the University of Chicago Facebook network, but are rather nested within them.

In section 4 we proposed an alternative evaluation scheme that is appropriate for the case where one has only an incomplete ground-truth, and is based on a classification task. While this evaluation scheme does handle the complications introduced by the nesting problem, it does not get around the problems associated with incompleteness. In practice, then, this evaluation scheme will only be able to measure an algorithm's recall. To measure precision, some other evaluation framework, such as can be carried out on synthetic benchmark graphs, should be carried out in tandem with the scheme proposed here.

Another unfortunate drawback of the benchmarking approach presented here is its complexity. The proposed benchmarking workflow involves components that are not directly related to community detection—such as classifiers—which add extra parameters whose values must be set with care by someone who is experienced with supervised machine learning. Also, because this classification-based evaluation has many moving pieces, in practice it will be hard to replicate independently. We therefore believe that if the research community is to use such classification-based evaluations, the source code used to carry them out should be shared in a public repository. This repository should be extensible so that new community detection algorithms can easily be added to it.

To demonstrate what such a repository would look like in practice, and to show that all of the difficulties mentioned here can be overcome, we have shared a publicly accessible repository that contains the source code of a classification-based evaluation using the Facebook100 dataset mentioned above. We have structured the evaluation routines as a set of easy-to-use command line tools in the hope that other researchers will be able to easily run this evaluation on their own community detection methods. **To reviewers: We will set up this repository if the paper is accepted (acceptance can of course be conditioned on us actually setting up the repository).**

In order to provide a concrete example of such classification-based evaluation, this repository also includes a series of experiments we have carried out which indicate that the Louvain method (a popular modularity-maximization algorithm [4]) and the InfoMap algorithm [29] fail perform poorly because the communities they detect are too coarse. The experiments also indicate that by using the parameterized version of modularity proposed in [7], smaller communities can be detected, which improves performance on the classification task.

We conclude by noting that while the evaluation framework proposed here was based on the task of inferring missing node attributes, we could construct conceptually similar benchmarks based on different tasks. One natural example would be to use network communities to perform supervised link prediction; this is a natural fit because presumably the processes responsible for link formation are closely related to the processes which form network communities. Another possibility would be to use network communities to compress the network: because communities are believed to account for many of the edges in a network (and a lack of community structure should indicate a sparse region of the network), they should also be relevant for lossless compression of a graph.

There are likely many such tasks for which one could make a case that community structure should be highly relevant. By compiling a collection of these tasks, and measuring how well community detection algorithms perform on the various tasks, we might find that some community detection algorithms are generally more useful for these tasks than others.

Acknowledgements

We thank Sune Lehmann for bringing Conrad Lee out to Copenhagen to discuss this project. We acknowledge Mason Porter's helpful suggestions and corrections. For providing the excellent Facebook100 dataset, we thank Amanda L. Traud, Peter J. Mucha, Mason A. Porter, Eric D. Kelsic, and

Adam D'Angelo. Further thanks are due to Colm Ryan for his advice on attribute imputation tasks, and to Aaron McDaid for helpful discussions and references.

Funding

This work is supported by Science Foundation Ireland under grant no. 08/SRC/I1407, Clique: Graph and Network Analysis Cluster.

References

[1] Ahn, Y., Bagrow, J. & Lehmann, S. (2010) Link communities reveal multiscale complexity in networks. *Nature*, **466**(7307), 761–764.

[2] Berriz, G. F., Beaver, J. E., Cenik, C., Tasan, M. & Roth, F. P. (2009) Next generation software for functional trend analysis. *Bioinformatics*, **25**(22), 3043–3044.

[3] Bishop, C. M. et al. (2006) *Pattern recognition and machine learning*, volume 1. springer New York.

[4] Blondel, V., Guillaume, J., Lambiotte, R. & Lefebvre, E. (2008) Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, **2008**(10), P10008.

[5] Clauset, A., Moore, C. & Newman, M. (2007) Structural inference of hierarchies in networks. *Statistical network analysis: models, issues, and new directions*, pages 1–13.

[6] Davis, A., Gardner, B. & Gardner, M. (1941) *Deep south*. University of Chicago Press Chicago.

[7] Delvenne, J., Yaliraki, S. & Barahona, M. (2010) Stability of graph communities across time scales. *Proceedings of the National Academy of Sciences*, **107**(29), 12755–12760.

[8] Duch, J. & Arenas, A. (2005) Community detection in complex networks using extremal optimization. *Physical review E*, **72**(2), 027104.

[9] Färber, I., Günnemann, S., Kriegel, H.-P., Kröger, P., Müller, E., Schubert, E., Seidl, T. & Zimek, A. (2010) On using class-labels in evaluation of clusterings. In *MultiClust: 1st International Workshop on Discovering, Summarizing and Using Multiple Clusterings Held in Conjunction with KDD*.

[10] Forsyth, E. & Katz, L. (1946) A Matrix Approach to the Analysis of Sociometric Data: Preliminary Report. *Sociometry*, **9**(4), pp. 340–347.

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

[12] Freeman, L. C. (2003) Finding social groups: A meta-analysis of the southern women data. In *Dynamic Social Network Modeling and Analysis. The National Academies*, pages 39–97. Press.

[13] Girvan, M. & Newman, M. (2002) Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, **99**(12), 7821–7826.

[14] Gregory, S. (2011) Fuzzy overlapping communities in networks. *Journal of Statistical Mechanics: Theory and Experiment*, **2011**(02), P02017.

[15] Lancichinetti, A. & Fortunato, S. (2009a) Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, **80**(1), 016118.

[16] Lancichinetti, A. & Fortunato, S. (2009b) Community detection algorithms: a comparative analysis. *Physical Review E*, **80**(5), 056117.

[17] Lancichinetti, A. & Fortunato, S. (2011) Limits of modularity maximization in community detection. *Physical Review E*, **84**(6), 066122.

[18] Lancichinetti, A., Fortunato, S. & Radicchi, F. (2008) Benchmark graphs for testing community detection algorithms. *Physical Review E*, **78**(4), 046110.

[19] Lancichinetti, A., Kivelä, M., Saramäki, J. & Fortunato, S. (2010) Characterizing the community structure of complex networks. *PLoS One*, **5**(8), e11976.

[20] Lee, C., Reid, F., McDaid, A. & Hurley, N. (2010) Detecting highly overlapping community structure by greedy clique expansion. *SNA-KDD 2010*, page 11.

[21] Lee, C., Reid, F., McDaid, A. & Hurley, N. (2011) Seeding for pervasively overlapping communities. *Physical Review E*, **83**(6), 066107.

[22] Leskovec, J., Lang, K. J., Dasgupta, A. & Mahoney, M. W. (2009) Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, **6**(1), 29–123.

[23] Marras, E., Travaglione, A., Chaurasia, G., Futschik, M. & Capobianco, E. (2010) Inferring modules from human protein interactome classes. *BMC systems biology*, **4**(1), 102.

[24] McDaid, A. & Hurley, N. (2010) Detecting highly overlapping communities with model-based overlapping seed expansion. In *Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on*, pages 112–119. IEEE.

[25] Moreno, J. (1934) *Who shall survive? : a new approach to the problem of human interrelations*. Nervous and Mental Disease Publishing Co.

[26] Newman, M. (2006) Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, **103**(23), 8577–8582.

[27] Porter, M. A., Onnela, J.-P. & Mucha, P. J. (2009) Communities in networks. *Notices of the AMS*, **56**(9), 1082–1097.

[28] Reid, F., McDaid, A. & Hurley, N. (2011) Partitioning breaks communities. In *Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on*, pages 102–109. IEEE.

[29] Rosvall, M. & Bergstrom, C. (2011) Multilevel compression of random walks on networks reveals hierarchical organization in large integrated systems. *PloS one*, **6**(4), e18209.

[30] Sampson, S. (1968) *A novitiate in a period of change: An experimental and case study of social relationships*. PhD thesis, Cornell University.

[31] Traud, A. L., Mucha, P. J. & Porter, M, A. (2012) Social structure of Facebook networks. *Physica A: Statistical Mechanics and its Applications*, **391**(16), 4165–4180.

[32] Watts, D. J. (2004) The" new" science of networks. *Annual review of sociology*, pages 243–270.

[33] Xie, J., Kelley, S. & Szymanski, B. K. (2011) Overlapping community detection in networks: the state of the art and comparative study. *arXiv preprint arXiv:1110.5813*.

[34] Yang, J. & Leskovec, J. (2012a) Defining and evaluating network communities based on ground-truth. In *Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics*, page 3. ACM.

[35] Yang, J. & Leskovec, J. (2012b) Structure and Overlaps of Communities in Networks. In *Proceedings of the 6th SNA-KDD Workshop*.

[36] Zachary, W. W. (1977) An Information Flow Model for Conflict and Fission in Small Groups. *Journal of Anthropological Research*, **33**(4), pp. 452–473.