Krasanakis *et al. Applied Network Science* (2020) 5:48
https://doi.org/10.1007/s41109-020-00287-x

Applied Network Science

RESEARCH Open Access

Unsupervised evaluation of multiple node ranks by reconstructing local structures

Emmanouil Krasanakis* , Symeon Papadopoulos and Yiannis Kompatsiaris

*Correspondence: maniospas@iti.gr
CERTH-ITI, Thessaloniki, Greece

Abstract

A problem that frequently occurs when mining complex networks is selecting algorithms with which to rank the relevance of nodes to metadata groups characterized by a small number of examples. The best algorithms are often found through experiments on labeled networks or unsupervised structural community quality measures. However, new networks could exhibit characteristics different from the labeled ones, whereas structural community quality measures favor dense congregations of nodes but not metadata groups spanning a wide breadth of the network. To avoid these shortcomings, in this work we propose using unsupervised measures that assess node rank quality across multiple metadata groups through their ability to reconstruct the local structures of network nodes; these are retrieved from the network and not assumed. Three types of local structures are explored: linked nodes, nodes up to two hops away and nodes forming triangles. We compare the resulting measures alongside unsupervised structural community quality ones to the AUC and NDCG of supervised evaluation in one synthetic and four real-world labelled networks. Our experiments suggest that our proposed local structure measures are often more accurate for unsupervised pairwise comparison of ranking algorithms, especially when few example nodes are provided. Furthermore, the ability to reconstruct the extended neighborhood, which we call HopAUC, manages to select a near-best among many ranking algorithms in most networks.

Keywords: Complex networks, Ranking algorithms, Metadata group communities, Evaluation measures

Introduction

The nodes of complex networks are often organized into (overlapping) communities that mirror the systemic properties of their real-world counterparts. Traditionally, researchers have theorized that this organization exhibits strong locality, i.e. that nodes with similar attributes are concentrated into small areas, and tried to discover structural ground truth communities whose nodes are tightly knit together (Fortunato and Hric 2016; Leskovec et al. 2010; Xie et al. 2013; Papadopoulos et al. 2012). However, recent works have found that node attribute values can also correlate to non-local structural features (Hric et al. 2016; Lancichinetti et al. 2009; Jeub et al. 2015), such as hierarchical dependencies. Thus, it has become clear that similarly-attributed nodes scattered throughout the network form a

different type of communities (Hric et al. 2014; Hric et al. 2016; Peel et al. 2017), referred to as *metadata groups* on merit that node attributes can be considered their metadata.

To understand the notion of metadata groups, these can be the product category in product co-purchasing networks, the preferences (e.g. genres of liked music) of social media users, or the scientific fields of publications in academic collaboration networks. Sometimes, metadata groups are organized into tightly knit structural communities. However, this happens only if they are correlated with the attributes most influencing the formation of network edges. Otherwise, they are not necessarily well-connected or clearly separated from the rest of the network. In fact, depending on the modeled attribute, metadata groups can be overlapping or span wide areas of a network, for example when social media users obtain multiple out of few available attributes.

The inclusion of network nodes in metadata groups is not always a binary property (De Domenico et al. 2015; Perer and Shneiderman 2006). For example, it is often of interest to rank the relatedness of social network users to an attribute, so as to avoid setting thresholds for systemically vague group boundaries (Lancichinetti et al. 2009; Leskovec et al. 2009). This is demonstrated in Fig. 1, where the nodes separating two communities (which could be metadata groups) pertain to both sides. Ranking nodes based on their relatedness to communities of interest is also a recurring task in the broader scope of mining complex networks. In particular, node ranks enable a more granular understanding of communities and can be used by recommender systems that combine them with other characteristics. In this case, it is important for ranks to be of high quality across the whole network. Furthermore, some of the most well-known algorithms that discover clear-cut communities given a few known members rely on ranking mechanisms and work by thresholding their outcome (Andersen et al. 2006; Whang et al. 2016; Wu et al. 2012). In some cases, the thresholded outcome can be used to train more sophisticated community detection algorithms, such as semi-supervised graph neural networks (Li et al. 2018; Kipf and Welling 2016), that need many training examples.

Node ranks for metadata groups are a form of recommendation and their quality can be (e.g. in Hsu et al. (2017)) evaluated with well-known recommender system measures (Shani and Gunawardana 2011; Wang et al. 2013; Isinkaye et al. 2015), such as AUC and NDCG. Such measures provide a fine-grained assessment of rank quality that reflects whether higher ranks pertain to higher relatedness to metadata groups. For example, when evaluating node ranks for the network of Fig. 1, they would assess as high quality ranks for the left group of Fig. 1 that identify nodes D, E as more related to it than F, G and nodes A, B, C as closer to its boundary. Other evaluation practices, such as thresholding

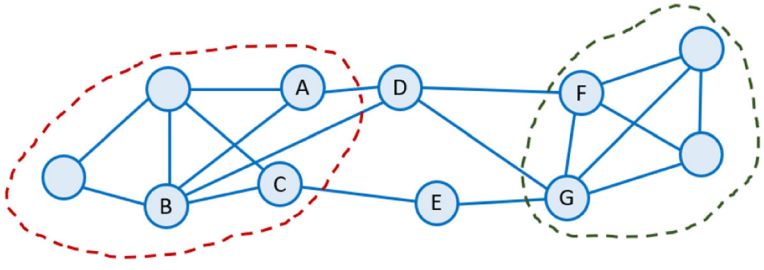

Fig. 1 Example of uncertain community boundaries, where two communities (circled) are equally well-connected to two boundary nodes D, E

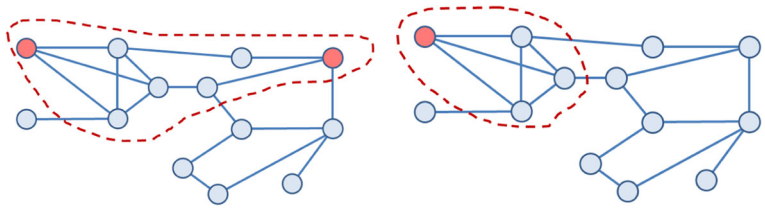

Fig. 2 Demonstration of underlying communities (circled) towards which to calculate node relevance given example nodes (highlighted). Removing one of the two example nodes heavily impacts the community we would want a ranking algorithm help suggest

node ranks and assessing the resulting clear-cut communities, can be too coarse in that they cannot necessarily tell whether ranks can help identify nodes related to metadata groups or whether ranks provide a granular understanding near group boundaries.

Since calculating recommender system measures requires knowledge of actual node labels, the efficacy of ranking algorithms is usually demonstrated on labeled networks, such as those of the SNAP repository (Stanford Network Analysis Project (SNAP) datasets 2009)[1] Even so, the best strategies and parameters for ranking nodes could depend on the network at hand. Furthermore, it has been previously reported (Andersen et al. 2006; Avrachenkov et al. 2018; Krasanakis et al. 2019b; Tan 2017) that ranking nodes in networks with large metadata groups (e.g. with many nodes and edges) requires different structural assumptions than in networks with smaller groups. This raises concerns over the ability of ranking algorithms tested on a few networks to work as well on different ones.

With this consideration in mind, we argue that node ranking algorithms and their parameters should be chosen anew for each network. However, large real-world networks are often sparsely labeled and all of the few example metadata group labels are needed to help extrapolate node ranks; otherwise, if example nodes are too few or scattered throughout the network, withholding some to calculate supervised measures could severely impact the quality of node ranks, as demonstrated in Fig. 2. Methodological concerns also arise when example nodes reside close to each other, for instance if they were annotated by human experts that looked only at a portion of the network and labeled the nodes found there. In this case, using the labeled nodes for evaluation may not capture the efficacy of ranking algorithms on the whole network.

If we cannot use supervised procedures to evaluate node ranks, then how about unsupervised ones? A first take on this would be to generalize traditional structural community measures, such as density (Kowalik 2006), modularity (Newman 2006) and conductance (Chalupa 2017), to support ranks. However, these measures are designed with structural ground truth communities in mind and thus do not account for non-local patterns that could characterize metadata groups (Hric et al. 2016; Lancichinetti et al. 2009; Jeub et al. 2015).

To create unsupervised procedures tailored to evaluating non-local metadata group communities, in our recent work (Krasanakis et al. 2019a) we proposed that only high-quality ranking algorithms can capture the relatedness of nodes to metadata groups that drive the formation of network edges. In particular, we adopted the assumption that

[1] The communities available of the SNAP repository are not metadata groups but partitions of those groups into subgroups of high conductance (Yang and Leskovec 2015).

network edges are often formed between nodes of similar metadata preferences (Hric et al. 2016), a phenomenon known as homophily in social networks (McPherson et al. 2001), and proposed assessing the quality of node ranks for multiple metadata groups through their ability to predict network edges. To do so, we used the similarity of rank distributions across groups to suggest links between nodes and evaluated those suggestions using AUC. We showed that this type of evaluation, which we called LinkAUC, enriches the concept of rank density and accounts for inter-group relations and experimentally asserted on a synthetic and two large real-world networks that it correlates to supervised AUC and NDCG more strongly than other rank-based measures when comparing conceptually different node ranking algorithms. A similar setting has also been independently proposed by Berry et al. (2020) for the inverse task of evaluating homophily given a high quality prediction of node attributes.

In this work, we extend several aspects of our previous research. First of all, we generalize the concept of predicting network edges to predicting other types of local structures that can be extracted from the network. This leads us to defining two new measures: HopAUC that captures the extended neighborhood up to two hops away, and LinkCC that compares the clustering coefficient between the original and reconstructed edges. We also introduce sampling strategies that allow fast computation of local structure evaluation in large networks. Thanks to this improved running time, we conduct more thorough experiments on a wider range of ranking algorithms and networks. To produce more ranking algorithms we experiment on a number of base algorithms, including those of our previous research, but this time we also explore perturbations of their parameters; finding the best combination of algorithms and parameters would be the target of optimization in industrial applications. In this context, we note that ranks obtained by small perturbations pertain to similar structural assumptions, which makes them more difficult to compare.

Our findings in this new series of experiments suggest that using local structures of the network to assess node ranks is a promising alternative to rank-based adaptations of structural community quality measures, where each network and number of example nodes warrant usage of different measures for comparing any two ranking algorithms. From a practical standpoint, we explore the more specific objective of using unsupervised measures to guide the ranking algorithm selection process towards a near-optimal algorithm. We find HopAUC to outperform other unsupervised node rank evaluation strategies in that respect.

Unsupervised evaluation of multiple node ranks

In the previous section we outlined the need for evaluating whether node ranks are indicative of their relatedness to metadata groups by using unsupervised procedures that do not necessarily favor densely-packed metadata group nodes. This way, we can select node ranking algorithms when the given example nodes are too few to conduct supervised experiments.

Approach motivation and overview

To develop unsupervised measures that account for non-local features, we argue that it is better to avoid making heuristic assumptions about high-level structural properties node ranks should exhibit, as these tend to favor local communities. Instead, ranks can

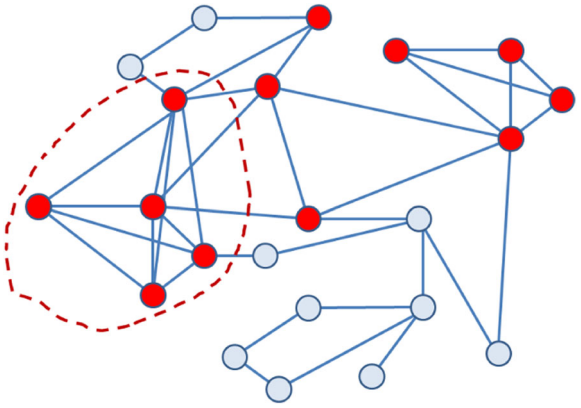

Fig. 3 Example community of red highlighted nodes that exhibit triadic closure (i.e. form triangles). The circled clique is tightly connected and hence would likely be favored by high level unsupervised structural community quality measures

be evaluated indirectly through their ability to reconstruct structural characteristics of known (as opposed to assumed) ground truth that exhibit similar quality to ranks. In particular, we consider local structures whose ground truth is readily available in the network.[2]

To understand why local structures could provide a more refined view of which nodes are accurately identified as members of metadata group communities, we point to Fig. 3, where the circled structural community comprises all 10 possible non-self-loop edges between its nodes and hence would be favored by the density measure later described in this section. However separating it from the network would remove the three triangles crossing its boundary that potentially link to more community members. In this example, high community density pertains to forming many triangles between its nodes. In the context of node ranks, this would translate to forming many triangles between similarly-ranked nodes. On the other hand, triangles do not reside only in the denser portions of the network, but also help link the circled clique on the left and the clique on the upper right. In a more general sense, structural community quality measures, such as density, can be considered high-level aggregations of local structures, such as triangles, but the inverse does not always hold true. This understanding suggests that structural community quality measures favor areas where many local structures reside, which at a higher level translates to community locality.

Contrary to the assumptions of structural community quality measures, the existence of many local structures in small areas could be influenced by other factors that are independent of the metadata group communities. For example, data gathering crawlers tend to exhaust all links between the first few found nodes but add edges with lower probability as more and more nodes are discovered. Bearing this problem in mind, we argue that the bias of local structure distribution across the network could be removed if, instead of aggregating the evaluation of local structures, we compared the structures predicted by node ranks against the structures actually present in the network.

[2] Local structures are not the same as communities that exhibit locality; the former refer to small-scale relations between nodes at most a few hops away, whereas the latter to congregations of many nodes.

In other words, we propose evaluating node ranks through their ability to predict local structures of the network. In our previous research we suggested using edges as the type of local structure to predict, since their existence represents the most fundamental type of structural information. However, here we also consider other types of local structures, such as the extended neighborhood, also known as "friends-of-friends", and triadic closure, which will be explained later on. These structures comprise specific edge combinations and hence our proposed evaluation requires a node comparison mechanism that transforms node ranks to link ranks of similar quality, i.e. that predict edges well only if node ranks mimic metadata groups and conversely, which can in turn be used to evaluate the examined type of local structures with similar quality. An overview of our proposed evaluation scheme is demonstrated in detail in Fig. 4, where the evaluation step at the end is responsible for assessing the quality of local structures arising from link ranks against the existing network edges.

To formally describe this scheme, we begin by annotating as r_i the vectors whose elements r_{ij} estimate the relevance of network nodes j to metadata groups $i = 1, \ldots, n$. Ranking algorithms often compute this relevance so that it resides in the interval $[0, 1]$, but it could assume any non-negative value as long as higher values indicate closer relatedness. For example, in the vector $r_i = [.1, .4, .6, .3]^T$ the third node should be the one estimated as the most closely related to the metadata group i. Each vector r_i holds the relevance of all nodes to one metadata group, but the same node may obtain non-zero relevance to multiple groups.

The intuition derived by latent factor models for link prediction (Hoff 2008; Menon and Elkan 2011; Duan et al. 2017) and collaborative filtering (Koren and Bell 2015) helps us recognize $R = [r_1 \ldots r_n]$ as a matrix factorization of the network, i.e. a representation of network nodes in a lower dimensional space where a similarity measure can reconstruct their structural relations. In detail, the rows $R_j = \left[r_{1j} \ldots r_{nj}\right]$ of this representation form distributions of ranks of network nodes j across all metadata groups. Then, following the principles of link prediction works (Liben-Nowell and Kleinberg 2007; Lü and Zhou 2011), if network construction is influenced predominantly by structure-based and metadata-based characteristics and these are captured by the node ranking algorithm, this factorization can help predict network edges by linking nodes with similar rank distributions. Conversely, if network edges are predicted with high quality by node ranks, then the ranking algorithm has captured the structure-based and metadata-based characteristics of the network.

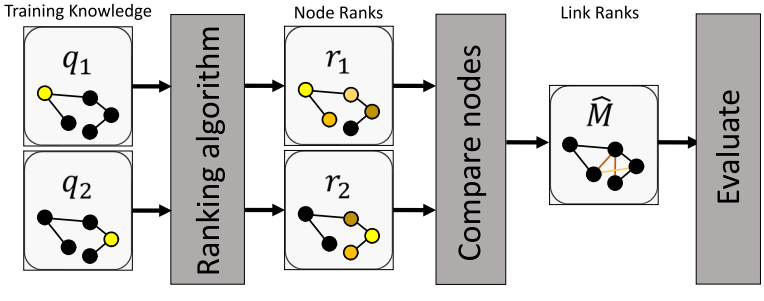

Fig. 4 Proposed scheme for evaluating ranking algorithms. Lighter colored nodes have higher ranks and are closely related to the respective metadata group. Lighter colored edges have lower link ranks are hence less likely to exist

To understand this claim, let us analyse the simple setting in which the similarities of rank distributions are calculated between nodes j, k as their dot product [3] $\widehat{M}_{jk} = R_j \cdot R_k$. The similarities of all node pairs can then be gathered in a link ranking matrix:

$$\widehat{M} = RR^T \tag{1}$$

Let us also consider ranking algorithms that can be expressed as network filters $f(M) = \sum_{n=0}^{\infty} a_n M^n$ (Ortega et al. 2018) of the (normalized) network's adjacency matrix M, where a_n are the weights placed on random walks of length n. When applied on query vectors q_i, whose elements q_{ij} are proportional to probabilities that nodes j belong to metadata groups i, network filters produce ranks $r_i = f(M)q_i$ of how much nodes pertain to the metadata groups. In the case of example nodes, $q_{ij} \in \{0, 1\}$, but elements of the query vectors can more broadly assume any non-negative value. Similarly to before, there may exist overlapping non-zero values between metadata group query vectors, for example if a node is a known member of multiple groups. Two well-known examples of network filters are personalized PageRank and Heat Kernels, which arise from exponentially degrading weights and the Taylor expansion coefficients of an exponential function respectively (see "Ranking algorithms" section for details).

Organizing multiple queries into a matrix $Q = [q_1 \dots q_n]$, we can express link ranks obtained by network filters as:

$$R = f(M)Q \Rightarrow \widehat{M} = f(M)QQ^T f^T(M) \tag{2}$$

This is a quadratic form of $f(M)$ around the kernel QQ^T and, as such, propagates the relations between example node pairs to the rest of link candidates. Therefore, if example nodes adequately capture the structural role of metadata groups and link ranks closely predict the network's edges, then the algorithm with filter $f(M)$ is a good rank propagation mechanism. At best, queries form an orthonormal basis of ranks $QQ^T = I$, which occurs only if each node is a known member of only one metadata group, i.e. if the ranking algorithm explores the relevance of network nodes to non-overlapping fully-known metadata groups. In this best case, link ranks assume the form $f(M)f^T(M)$, which can model the whole space of symmetric link prediction filters (Liben-Nowell and Kleinberg 2007; Lü and Zhou 2011; Martínez et al. 2017). Therefore, the quality of $f(M)$ is directly tied to the ability of the ability of the filter $f(M)f^T(M)$ to reconstruct network edges.

The above analysis motivates the development of new unsupervised strategies for evaluating node ranks through their link predictive capabilities. In their most general form, such strategies comprise an unsupervised evaluation measure of link rank quality $u(\widehat{M}, M)$ that is maximized as $\widehat{M}$ approaches M.

In this section we detail three strategies of this type, which we call LinkAUC, HopAUC and LinkCC, that evaluate link ranks through the concepts of edge prediction, extended neighborhood prediction and triadic closure. We also provide rank-based adaptations of existing structural community quality measures, on merit that they mirror existing practices widespread in the literature, which assume that metadata groups pertain to structural communities. An overview of all measures outlined in this section is provided in Table 1. In that table, the running time required to compute each measure is denoted through the big-O notation as a function of the number of nodes N and edges E of the

[3] Cosine similarity would arise by a fixed-flow assumption of the ranking algorithm that performs row-wise normalization of R before the dot product.

Table 1 Proposed measures to assess node rank quality

Measure		Objective Type	Time Complexity
Density	Kowalik (2006)	Structural communities	$O(E)$
Conductance	Chalupa (2017)	Structural communities	$O(E)$
Modularity	Newman (2006)	Structural communities	$O(N^2)$
LinkAUC	Krasanakis et al. (2019a)	Local structures	$O(N^2)$
HopAUC	[This work]	Local structures	$O(N^2)$
LinkCC	[This work]	Local structures	$O(E^2/N)$

network. In "Scaling local structure evaluation to large networks" section we introduce scalable sampling procedures that significantly reduce the running time of the slower local structure measures in large networks.

Network groups

Before moving on, we point out that evaluating link ranks often requires exclusion of certain links, such as withheld test edges, those absent due to systemic reasons (e.g. users may not be allowed to befriend themselves in social networks) or those sampled out of the evaluation to scale down the time needed to approximate the measures in large networks. To model the absence of links in the same way throughout all measure definitions, we introduce the idea of using a binary matrix $\mathbb{M}$ to remove non-comparable links of the network's adjacency matrix M.

To do this, the adjacency matrix can be projected to $M \odot \mathbb{M}$, where $\odot$ is the multiplication of corresponding elements, also known as the Hadamard product. For example, removing zero-diagonal networks corresponds to $\mathbb{M} = \mathbf{1} - I$, where $\mathbf{1}$ are square matrices of ones and I identity matrices with the same dimensions as M. This projection process retains closure, i.e. $M_1 \odot \mathbb{M} + M_2 \odot \mathbb{M} = (M_1 + M_2) \odot \mathbb{M}$, and matrix addition properties, such as having a negation $-(M \odot \mathbb{M}) = (-M) \odot \mathbb{M}$. Hence, the outcome of this operation for the binary matrix $\mathbb{M}$ forms an algebraic group with regards to addition and we will call it *network group* (not to be confused with the concept of metadata groups).

To use network groups in evaluation, both the network's adjacency matrix and estimated link ranks need be projected into the same network group designated for evaluation. An example of the projection process is demonstrated in Fig. 5. As an additional helpful tool to facilitate formal representation of this projection when element-by-element comparisons are needed, we introduce a transformation $vec_{\mathbb{M}}(M)$ that creates a vector containing all elements of M for which $\mathbb{M} \neq 0$ in a predetermined order. For example, we could consider that $vec_{[0,1;1,0]}([a, b; c, d]) = [b, c]$.

Adapting structural community quality measures to node ranks

We have already mentioned that a first take to enabling unsupervised evaluation of node ranks would be adapting existing structural community quality measures to account for them. To this end, one could replace quantities used to calculate community properties, such as their number of internal edges, with their expected value across the fuzzy set of subgraphs arising from ranks being proportional to the probabilities that nodes are members of found communities.

In particular, if we consider the node rank vector r_i, where r_{ij} follows our previous notation and shows how much nodes j pertain to the community this vector refers to,

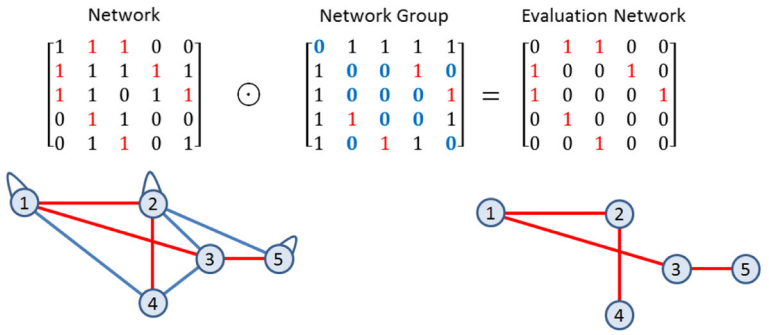

Fig. 5 A network (left) projected for evaluation (right) to a network group (middle) that removes the evaluation of the blue edges. Rows and columns of the adjacency matrices correspond to the respective node numbers

we can define a vector sampling strategy V_i whose elements V_{ij} are binary random variables that select whether nodes j have been selected as members of the community with probabilities proportional to r_{ij}. Formally, these random variables can be expressed as:

$$V_{ij} = \left\{1 \text{ with probability } \frac{r_{ij}}{\|r_i\|_1}, 0 \text{ otherwise}\right\}$$

where $\|\cdot\|_1$ is the L1 norm, calculated as the sum of vector elements, and v are binary vectors of vertices sampled with probabilities r. Then, the fuzzy community $\mathbb{I}$ corresponding to the rank vector r_i can be defined as the value set of obtaining sampled vectors $v_i \sim V_i$ arbitrarily many times, i.e. $\mathbb{I} = \{v_i : v_i \sim V_i\}$. For the sake of brevity, we refrain from explicitly redefining the random variable each time and annotate the fuzzy community as $v_i \sim r_i$.

Given this setup, any community-specific measure $f(M, v)$ on a network adjacency matrix M that was originally defined only for binary vectors v whose elements reflected whether nodes belong to a community or not can be generalized to node ranks r_i by obtaining its expected value over the fuzzy set of communities:

$$f(M, r_i) = \mathbb{E}_{v_i \sim r_i}\left[f(M, v_i)\right] \tag{3}$$

Under this formulation, if node ranks were to assume binary values (e.g. because they indicate community memberships), the fuzzy community would consist of only one element that would correspond to its non-stochastic counterpart. Using this method to account for ranks, we extend three of the most popular structural community quality measures; density, conductance and modularity.

Density. The density of a network is defined as the portion of its edges compared to the maximal number of possible ones (Kowalik 2006; Schaeffer 2007; Görke et al. 2015). Then, the density of communities is defined as the density of the sub-network comprising the edges between community nodes. Using the notion of volume $vol(M)$ to annotate the number of edges in a network with adjacency matrix M, the density of its projection inside the network group $\mathbb{M}$ becomes $D_{\mathbb{M}}(M) = \frac{vol(M \odot \mathbb{M})}{vol(\mathbb{M})}$. We similarly define rank density by substituting the volume with the expected volume $vol(M, r)$ of the aforementioned fuzzy

set of subgraphs arising from node ranks:

$$\begin{aligned} vol(M, r) &= \mathbb{E}_{v\sim r}\left[v^T M v\right] = \frac{r^T M r}{\|r\|_1^2} \\ \Rightarrow D_{\mathbb{M}}(M, r) &= \frac{vol(M \odot \mathbb{M}, r)}{vol(\mathbb{M}, r)} = \frac{r^T (M \odot \mathbb{M}) r}{r^T \mathbb{M} r} \end{aligned} \tag{4}$$

Conductance. Conductance compares the number of links leaving a community with the number of its internal links (Chalupa 2017). Hence, it reflects the probability of a process that randomly walks the network to move outside that community vs. the probability to remain inside it (Andersen et al. 2008). Using the same probabilistic formulation as for rank density, rank conductance can be defined as:

$$\phi_{\mathbb{M}}(M, r) = \frac{r^T (M \odot \mathbb{M})(C - r)}{r^T (M \odot \mathbb{M}) r} \tag{5}$$

where C is a max-probability parameter. Since comparisons are preserved for any value of this parameter, we arbitrarily select $C = 1$. Lower conductance indicates better separation between nodes found to pertain to metadata groups and the rest of the network.

Modularity. Modularity measures the number of edges within the same community against the number of edges that would be obtained if nodes were linked at random to maintain their degrees (Newman 2006). This comparison can be written as $Q(M) = \frac{1}{2m}\sum_{v,u\in c}\left(M_{vu} - \frac{D_{vv}D_{uu}}{2m}\right)$ where D_{vv} represents the degree of node v, m is the number of network edges and c is the found community. To constraint this search in a network group $\mathbb{M}$, we can aggregate the link evaluation weight $A_{vu} = M_{vu} - \frac{D_{vv}D_{uu}}{2m}$ only across the links (v, u) allowed in the network group. Then, replacing the found community with the same fuzzy definition as before yields the following expectation of modularity for node ranks r:

$$Q_{\mathbb{M}}(M, r) = \frac{1}{2m} r^T (A \odot \mathbb{M}) r \tag{6}$$

where $A = M - \left[\frac{D_{vv}D_{uu}}{2m}\right]_{v,u}$.

Local structure evaluation of node ranks

The previous node rank measures rely on node ranks satisfying different structural characteristics across a fuzzy understanding of communities. However, we have argued that assessing node ranks through their ability to predict the local structures of networks can also provide meaningful insights towards the quality of given ranks. In this work we explore structures of the lowest levels; edges, extended neighborhoods and triangles. In our previous work, we already provided a measure for assessing edge prediction, which we called LinkAUC. For the sake of completeness, we present this measure again. We then provide a similar link prediction scheme that tries to reconstruct the extended neighborhood up to two hops away from each node, which we name HopAUC. Finally, we evaluate triangles through a measure we call LinkCC that compares a stochastic adaptation of the clustering coefficient with the clustering coefficient exhibited by the network.

LinkAUC. Evaluating link ranks $\widehat{M}$ against a network with adjacency matrix M within a network group $\mathbb{M}$ can be considered equivalent to comparing the corresponding potential links of the vectors $vec_{\mathbb{M}}(\widehat{M})$ with $vec_{\mathbb{M}}(M)$. A robust measure that compares operating

characteristic trade-offs of ranking mechanisms at different decision thresholds is the Area Under Curve (AUC) (Hanley and McNeil 1982). This measure has also been previously used to evaluate link ranks (Lü and Zhou 2011). When network edges are not weighted, if $TPR(\theta)$ and $FPR(\theta)$ are the true positive and false positive rates of a decision threshold θ on $vec_{\mathbb{M}}(\widehat{M})$ predicting $vec_{\mathbb{M}}(M)$, the AUC of link ranks becomes:

$$LinkAUC = \int_{-\infty}^{\infty} TPR(\theta)FPR'(\theta)\, d\theta \tag{7}$$

This evaluates whether actual linkage is assigned higher ranks across the network (Mason and Graham 2002) without being affected by edge sparsity. These properties make LinkAUC preferable to precision-based evaluation of link ranks, which assesses the correctness of only a fixed number of top predictions (Lü and Zhou 2011).

An interesting property of LinkAUC is that it can be considered an enrichment of rank density. For example, if we first examine the qualitative relation between link ranks and rank density for a single metadata group $R = r_1$. Annotating as $m \geq \theta$ the vectors arising from binary thresholding on the elements of $m = \frac{vec_{\mathbb{M}}(\widehat{M})}{\|vec_{\mathbb{M}}(\widehat{M})\|_1}$ and selecting thresholds $\theta[k]$ that determine the top-k link ranks up to all K link candidates ($\theta[K] = 0$):

$$m = \sum_{k=1}^{K-1} (m \geq \theta[k])(\theta[k] - \theta[k+1])$$

$$\Rightarrow D_{\mathbb{M}}(M, r_1) = \frac{vec_{\mathbb{M}}^T(M) vec_{\mathbb{M}}(\widehat{M})}{\|vec_{\mathbb{M}}(\widehat{M})\|_1} = \int_{-\infty}^{\infty} TP(\theta)P'(\theta)d\theta$$

where TP and P denote the number of true positive and positive number of thresholded link ranks respectively. At worst, every new positive link after a certain point would be a false positive. Using the big-O notation this can be written as $\frac{\partial FPR(\theta)}{\partial P(\theta)} \in O(1)$ and hence:

$$LinkAUC \in O\left(D_{\mathbb{M}}(M, r_1)\right) \tag{8}$$

Then, if we consider the case where discovered ranks form non-overlapping metadata groups, i.e. each node has non-zero rank only for one group. This may happen when query propagation stops before it reaches other metadata groups. Annotating $\widehat{M}_i = r_i r_i^T$, for non-overlapping ranks $r_i \cdot r_j = 0$ for $i \neq j$, we rewrite (1) as $\widehat{M} = \sum_i \widehat{M}_i \Rightarrow vec_{\mathbb{M}}(\widehat{M}) = \sum_i vec_{\mathbb{M}}(\widehat{M}_i)$, similarly to before:

$$LinkAUC \in O\left(\sum_i D_{\mathbb{M}}(M, r_i) vol(\mathbb{M}, r_i) \|r_i\|_1^2\right)$$

This averages group densities and weighs them by $vol(\mathbb{M}, r_i)\|r_i\|_1^2$. Hence, when metadata groups are non-overlapping, high LinkAUC indicates high rank density.

Finally, for overlapping metadata groups, LinkAUC involves inter-group links in its evaluation. Since averaging density-based evaluations across groups ignores these links, LinkAUC can be considered an enrichment of rank density in the sense that it bounds it when metadata groups do not overlap but accounts for more information when they do.

HopAUC. LinkAUC evaluates whether the distribution of node ranks across multiple metadata groups can reconstruct the immediate neighbors of nodes. However, it can be argued that higher-order notions of node proximity, i.e. of nodes laying more than one hop away from each other, also capture important structural aspects of the graph

that merit reconstruction, for example in the context of extracting low-dimensional node embeddings (Tang et al. 2015; Wang et al. 2016; Yang et al. 2017).

Second-order proximity in particular, which accounts for the neighbors of neighbors, has been found to naturally arise in real-world networks (Jin et al. 2001; Dash 2018; Tang et al. 2015). Therefore, if we consider node rank distributions across metadata groups to be low dimensional representations of the nodes' structural roles, these could potentially be evaluated by this type of proximity. To this end, we propose enriching LinkAUC by extending each node's notion of neighborhood to the extended node neighborhood comprising nodes up to two hops away. For example, in Fig. 6 this enrichment would involve checking whether nodes lying in the circled extended neighborhood of the red highlighted ones exhibit greater similarity with the latter compared to nodes lying outside each other's neighborhood. In this particular graph, this means that more emphasis is placed on predicting the dissimilarity with nodes farther away than on the exact node order paths of up to 2 edges follow.

To capture this type of local structure in a network group $\mathbb{M}$ for a binary network adjacency matrix M, we use a thresholding operation $\geq$ to define a measure we call *HopAUC* that calculates the AUC of $vec_{\mathbb{M}}(\widehat{M})$ compared to the ground truth extended neighbor $vec_{\mathbb{M}}(M + M^2 \geq 1)$.

LinkCC. The clustering coefficient (Wu et al. 2016; Opsahl and Panzarasa 2009) is an unsupervised measure of how well-connected neighbors are in a network, a concept also known as triadic closure. The clustering coefficient of unweighted networks is calculated as the number of triangles (i.e. triples of nodes linked with each other) that are formed between nodes over the maximal possible number of such triangles that would occur all the neighbors of network node neighbors linked back to the starting nodes.

If we assume that link ranks $\widehat{M}$ model the probability of the respective edges existing, we can use the weighted definition of the clustering coefficient as the expected number of triangles compared to the expected number of triangles if edges adhered to triadic closure. This yields to expressing it as $CC(\widehat{M}) = \frac{tr(\widehat{M}^3)}{tr(\widehat{M}\mathbf{1}\widehat{M})}$, where $tr(\cdot)$ is the trace operator retrieving the sum of a matrix's diagonal elements and $\mathbf{1}$ denotes a square matrix of ones. An intuitive understanding of this formula's nominator is that $\widehat{M}^3$ reflects the graph operation of moving three hops away with probability proportional to link ranks and summing

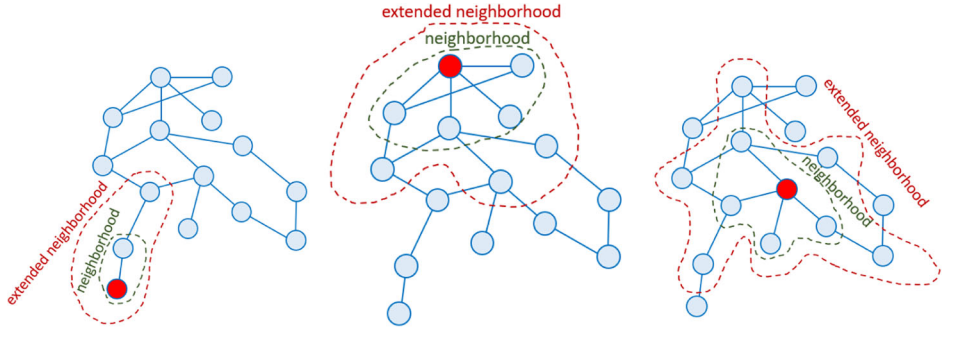

Fig. 6 The neighborhoods and extended neighborhoods of three red highlighted nodes in a toy network. LinkAUC assesses node ranks by their ability to predict which nodes are in each other's neighborhoods, whereas HopAUC performs the same assessment for the extended neighborhoods

the diagonal elements represents counting the expected number of those hops returning to the starting node. Similarly, the denominator $\widehat{M}\mathbf{1}\widehat{M}$ represents the operation of hopping to a neighbor then to any node and then to a neighbor of the last node.

Unfortunately, most elements of the link rank matrix are non-zero (even if some are very small). This explodes the number of computations needed to calculate its clustering coefficient to N^6 multiplications, where N is the number of network nodes; even the scalable approach of the next subsection would be hard-pressed to scale this computational cost to large networks. To avoid this issue, we propose a heuristic adaptation of the clustering coefficient, that instead of counting the expected number of triangles for link ranks, counts the expected number of triangles *given* the ground truth that two of their edges linking each node to its neigbhors already exist:

$$aCC(\widehat{M}, M) = \frac{tr\left(M\widehat{M}M\right)}{tr\left(M\mathbf{1}M\right)}$$

Then, we go back to our goal of comparing link ranks with the network's ground truth. aCC does involve both link ranks and the network's ground truth. However, it is not necessarily maximized when link rank prediction is of high quality. For example, if link ranks predict the network's edges near-perfectly, $aCC(\widehat{M}, M)$ could still assume small values if the clustering coefficient $CC(M) = aCC(M, M)$ of the network is small. To tackle this problem, we normalize the adapted definition of the clustering coefficient with the clustering coefficient of the network:

$$LinkCC(\widehat{M}, M) = \frac{aCC(\widehat{M}, M)}{aCC(M, M)} = \frac{tr\left(M\widehat{M}M\right)}{tr\left(M^3\right)} \tag{9}$$

As an example to understand our proposed adaptation for evaluating the triadic closure of link ranks, in Fig. 7 we show this evaluation would be conducted in an example network. From the viewpoint of node A the same number of triangles $|\{BD, BC, DE\}| = |\{BC, DE, CE\}|$ are predicted as in the original network, even if this number arises from predicting edges other than the actual ones. From the viewpoint of E half of the triangles are predicted $|\{AC\}| = 0.5|\{AC, DA\}|$. Repeating this process for the rest of the nodes, these evaluations are finally aggregated across by summing the total number of found closures and dividing with the total number of existing closures (e.g. if we accounted only for A and E this would be $(3 + 1)/(3 + 2) = 0.8$).

Scaling local structure evaluation to large networks

Table 1 showcases the computational time needed to calculate the explored measures on a given network. Measures that require $O(E)$ time can be considered to scale well with the size of the network, as the resources needed to rank nodes given E edges require at least

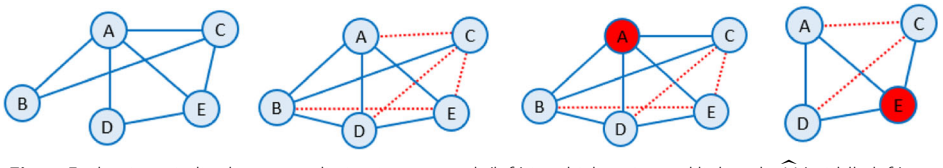

Fig. 7 Evaluating triadic closure prediction in a network (left) in which estimated link ranks $\widehat{M}$ (middle left) assume near-zero values for the dashed links and near-one for the blue links. LinkCC over the neighborhood of nodes A (middle right) and E (right) assumes that immediate neighbors are known and evaluates how many triangles are formed between them compared to the number of triangles in the original network

that much time to account for all dependencies between network nodes in connected networks. On the other hand, measures that require times $O(N^2)$ by comparing all potential links with the existence of edges can be prohibitively expensive to run when the number of network nodes N is large (e.g. millions). [4]. To calculate these measures in a reasonable time, we adopt sampling strategies that limit the number of links involved in evaluations. These strategies can be expressed as network groups by identifying the (binary) matrix $\mathbb{M}$ of sampled potential link pairs as the network group in which to calculate measures.

In our previous work, we used a uniform sampling strategy to select a number of test network nodes and checked their links with all other network nodes. This reduced the running time of checking all link pairs to $O(N_eN)$, where N_e is the number of example nodes and N the number of network nodes. Unfortunately, selecting too few nodes for evaluation distorts evaluation outcome by biasing towards the specific structural characteristics of those nodes. For example, there exist concerns over the robustness of identifying erroneous examples when evaluation nodes number up to the square root of all network nodes (Keith Borland 1950).

Given that there exists a theoretical lower limit for N_e, we propose reducing the running time of slower measures by also limiting the number of potential links they are evaluated on. To this end, we borrow the well-known negative sampling procedure (Goldberg and Levy 2014; Levy and Goldberg 2014), which reduces the number of computations needed to compute loss functions by uniformly sampling among the potential negative examples. In our case, negative examples constitute missing network links.

To formally express the combination of the above sampling strategies for evaluating link ranks against the network's adjacency matrix M (in the case of HopAUC, this would be the extended neighborhood adjacency matrix) we construct the binary queries q_e of sampled nodes, for which $q_e[j] = 1$ only if j is one of the N_e sampled for evaluation, and q_{neg} of similarly sampled nodes used to construct negative example links. Then, those queries can be used to define the network group $\mathbb{M} = \left(q_eM\overrightarrow{1}^T + q_eq_{neg}^T \geq 1\right)$, where $\overrightarrow{1}$ is a vector of ones, i.e. $\overrightarrow{1}[j] = 1$ for all nodes j. This network group comprises the edges between nodes of q_e and their network neighbors, as well as the potential links between q_e and q_{neg}.

If we sample N_{neg} negative nodes, and given that we fully omit operations involving potential links outside the network groups, the time needed to compute measures whose original complexity was $O(N^2)$ is then reduced to $O(N_eE/N + N_eN_{neg})$. Since $N_e \leq N$, this time is of order $O(E + N_eN_{neg})$. Hence, it can be considered to scale well in large networks if we choose to scale the number of positive and negative samples so that $N_eN_{neg} \in O(E)$.

Experiment setup

To assess the merit of evaluating node ranks using the local structure measures proposed in this work, we conduct a series of experiments on labeled networks that compare unsupervised and supervised measures across a wide range of potential ranks. Our goal is to identify which of the unsupervised measures best mimic the evaluation of the supervised ones, so as to use them in new networks, where supervised evaluation may not be applicable. We propose the following plan to tackle this type of meta-assessment:

[4] Real world networks tend to be extremely sparse, i.e. $E \ll N^2$

1. Create an experiment setting by selecting a network with known metadata groups and splitting the latter into example and evaluation nodes. For example, in a network with six nodes and two metadata groups that span the first four and last four nodes respectively, we can organize the known metadata group memberships into a query matrix $Q^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$, which can be randomly split into training and evaluation query matrices respectively as $Q^T_{train} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ and $Q^T_{evaluation} = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$ where each row of the transposed query matrices corresponds to a metadata group and each column to a node.
2. Run a variety of node ranking algorithms that start from the example nodes and rank the rest of network nodes based on their relevance to the metadata group of the examples. This yields a collection of node ranks for each algorithm that comprises the ranks for all metadata groups of the experiment setting. For example, the training queries Q^T_{train} of the previous step could produce node ranks $R^T_A = \begin{bmatrix} .4 & .3 & .2 & .1 & 0 & .1 \\ 0 & 0 & 0 & .2 & .2 & .6 \end{bmatrix}$ and $R^T_B = \begin{bmatrix} .3 & .3 & .2 & 0 & .1 & 0 \\ .2 & .1 & .2 & .3 & .2 & .1 \end{bmatrix}$ when inserted to algorithms A and B respectively, where each row of the transposed ranks corresponds to node ranks for the respective metadata group.
3. Use all examined unsupervised and supervised measures to evaluate the quality of each algorithm. This process yields a list of scores that summarize each algorithm's efficacy under the proposed measures. It must be noted that, at this point, we do not aim to find highly-scored algorithms but rather to find the unsupervised measures whose scores are indicative of node rank quality. For example, a measure u could use R_A and R_B to provide assessments $[u(A), u(B)] = [70\%, 60\%]$, whereas another measure v could provide assessments $[u(A), u(B)] = [80\%, 90\%]$.
4. Compare the unsupervised with the supervised measures to see which of the unsupervised ones are most suitable to identifying the best algorithms. For example, if the AUC scores for R_A and R_B averaged across metadata groups were $[AUC(A), AUC(B)] = [80\%, 70\%]$, u would be considered superior to v in selecting the ranks of higher quality.

These steps are visually demonstrated in Fig. 8 for a toy selection of algorithms and measures to compare. Our experiments are conducted on a wider scale, in which we set up a total of 72 ranking algorithms, each comprising a different base algorithm, parameters and strategy to augment its outcome by suggesting additional example nodes. These algorithms are also run on 25 experiment settings, which result from combining five different labeled networks with using different fractions of their known metadata group labels as examples. Hence, we compare measures over a total of $72 \times 25 = 1800$ group rank collections. In this section we explain the details of our methodology.

Experiment settings

Experiments are run on five networks; a synthetic one constructed through a stochastic block model (Holland et al. 1983), the large *Amazon* co-purchasing (Leskovec et al. 2007) and the *DBLP* author co-authorship networks that are often used to evaluate metadata

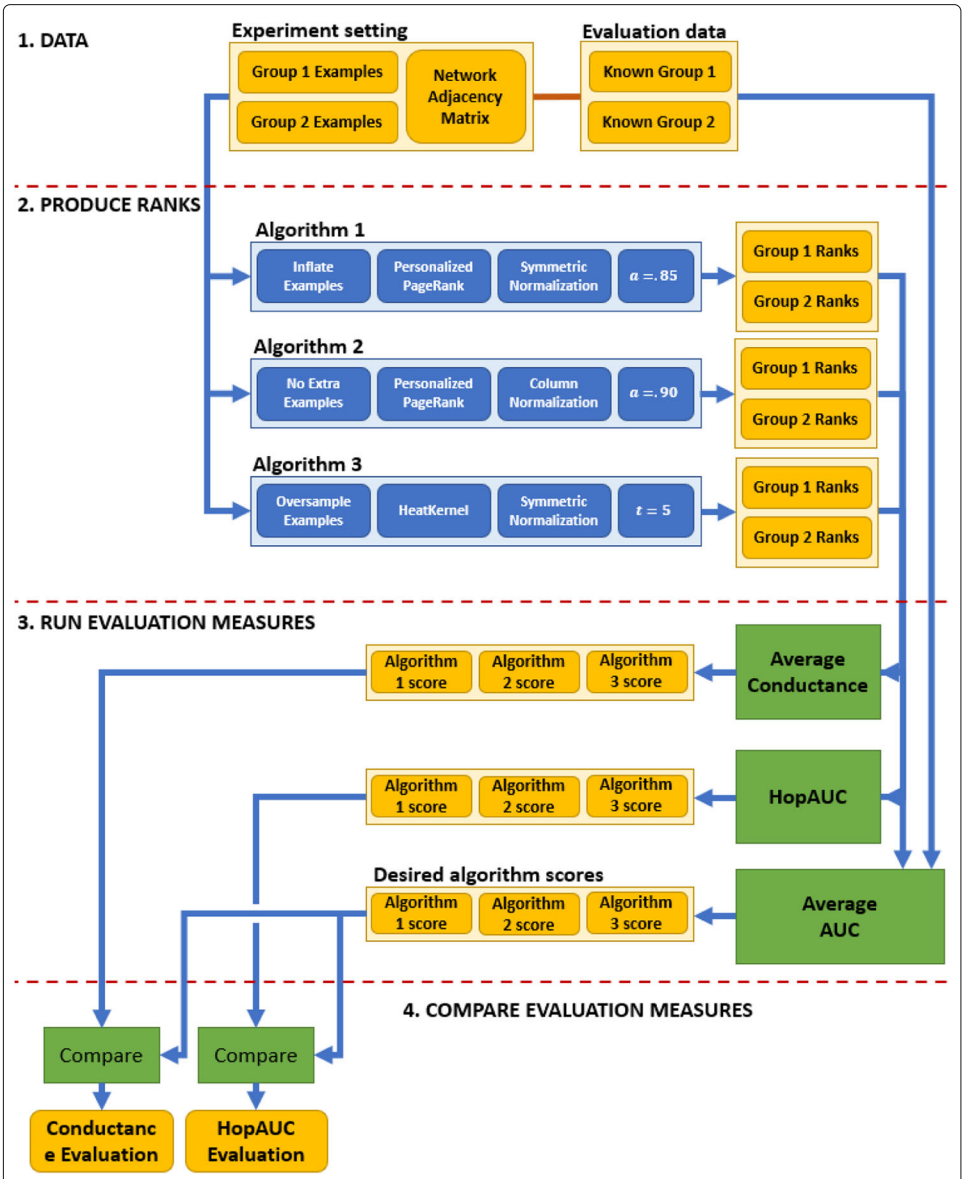

Fig. 8 Example of evaluating the ability of two unsupervised node rank quality measures (conductance and HopAUC) to compare three ranking algorithms in the same way as supervised AUC does. As a final step, we need to select the best of the unsupervised measures. Experiments in this work span 6 unsupervised evaluation measures, 72 ranking algorithms, are conducted for 25 experiments settings -which involve multiple metadata groups- and use both AUC and DCG for supervised comparison

group detection and the smaller *CiteSeer* (Sen et al. 2008) and *PubMed* (Namata et al. 2012) citation networks. All networks were selected on merit of their nodes being labeled, hence enabling supervised evaluation to serve as ground truth. They also comprise multiple metadata groups needed for link-based measures. Finally, although some of them (i.e. the citation networks) are directed, we convert them to undirected ones by considering nodes to be linked if an edge exists in either direction between them.

The stochastic block model is a popular method for constructing networks of known communities (Rohe et al. 2011; Abbe et al. 2016), where the probability of two nodes being linked is determined by which communities they belong to. Our synthetic network

uses the randomly generated 5×5 block probability matrix of Fig. 9 with blocks of 2K-5K nodes. The Amazon network comprises links between frequently co-purchased products (Amazon product co-purchasing network metadata 2007) that form communities based on their type (e.g. Book, CD, DVD, Video). We use the 2011 version of the DBLP dataset (DBLP Citation network 2011), which comprises 1.6M papers from the DBLP database, from which we extracted an author network based on co-authorship relations. In this network, authors form overlapping metadata groups based on academic venues (journals, conferences) they have published in. Then, CiteSeer (CiteSeer network 2003) is a small network that contains 3K publications each assigned one out of six labels and roughly 5K links of which publication cited each other. Finally, the PubMed network (PubMed network 2012) contains 20K medical publications labeled according to the type of diabetes they examine and 44K references between them.

A summary of the above networks is presented in Table 2. To limit the running time of our experiments, we use only the four and six largest metadata groups of the Amazon and DBLP networks respectively. In the last column we measure homophily as the fraction of edges linking nodes of the same metadata groups involved in experiments. It must be noted that this measure is at best a lower bound of how much similar network nodes tend to link to each other (a more detailed discussion on observed vs. actual homophily can be found in "Dynamics of local structures" section).

Given these networks, experiment settings are constructed by selecting a percentage among {0.1%, 1%, 10%, 25%, 50%} and gathering that many example nodes out of the nodes belonging to each metadata group, retrieving at least one example for each group. In total, combining the experimented networks and numbers of metadata groups yields $5 \cdot 5 = 25$ settings.

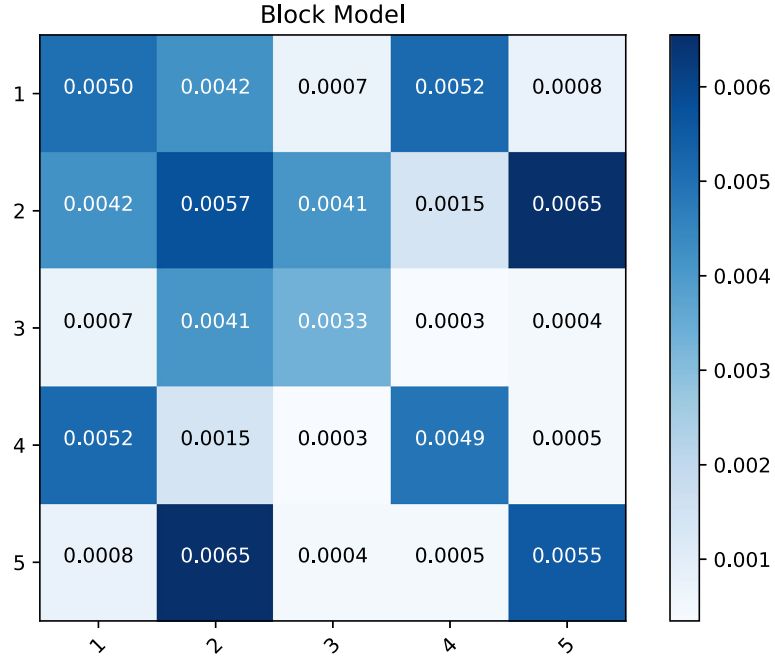

Fig. 9 The stochastic block model used to create the synthetic network. We synthesize edges between communities using the probabilities found in the respective cells

Table 2 Networks and the number of metada groups used in experiments

Network		Nodes	Edges	Groups	Homophily
Synthetic		15K	0.4M	5	52%
Amazon	Leskovec et al. (2007)	0.5M	1.8M	4	61%
DBLP	Tang et al. (2008)	1.0M	11.3M	6	10%
CiteSeer	Sen et al. (2008)	4K	5K	6	74%
PubMed	Namata et al. (2012)	20K	44K	3	80%

It could be argued that unsupervised evaluation is unnecessary for many (e.g. 50% of total) known example nodes, since there are enough to split into training and test sets without severely impacting the assessment quality. However, presenting the outcome of unsupervised measures in these cases can help us gain additional insights over their applicability and shortcomings.

Ranking algorithms

In this subsection we describe the ranking algorithms used to procure node ranks. Our objective is not to extract the best algorithm but for node rank quality measures to be computed on algorithms of varying efficacy, so as to find which unsupervised measures best match supervised ones.

For the sake of producing algorithms of varying efficacy, we construct them by choosing one of three base algorithms, the well-known personalized PageRank and Heat Kernels and a heuristic combination of their assumptions, and combining those four different values of their parameters that affect the spread of ranks, two different kinds of adjacency matrix normalization (symmetric or column-based) and three methodologies for enriching example nodes with more likely examples; no augmentation, inflation of the example nodes and oversampling of the example nodes. These combinations yield the aforementioned total of $3 \cdot 4 \cdot 2 \cdot 3 = 72$ different ranking algorithms. We expect some of the ranking algorithms relying on PageRank and Heat Kernels to produce ranks of high quality, whereas some of those that rely on the heuristic to be of lower quality. This way, we produce ranks with a wide range of qualities.

Personalized PageRank. Personalized PageRank (Andersen et al. 2006; Lofgren et al. 2016) is graph filter that arises from a random walk with restart strategy. In detail, it models a random process that starts from an example node and at each step either visits a random neighbor or restarts from a new random example with probability $1 - a$. This scheme for finding the ranks of a metadata group is traditionally expressed through the formula $r = aMD^{-1}r + (1 - a)q$, where q is the query vector of example nodes and D is the diagonal matrix of node degrees used to normalize the network's adjacency matrix M. In our experiments, we use the power method for computing personalized PageRanks, which iterates this formula until the mean absolute change of r_i becomes less than 10^{-6}. The parameter a, also called a dampening factor, determines how far random walks extend from the example nodes. Given previous guidelines about this parameter's impact on ranking metadata group nodes, we experiment with the values $a \in \{0.85, 0.90, 0.95, 0.99\}$.

An adjustment to personalized PageRank that is often considered is normalizing the adjacency matrix in a symmetric manner, for example to capture the undirectional nature of relations or to make ranks satisfy spectral graph analysis properties. To generalize the

previous formula to any kind of adjacency matrix normalization, we can express it as:

$$r = aW\frac{r}{\|r\|_1} + (1-a)\frac{q}{\|q\|_1} \tag{10}$$

where vector normalization with the L1 norm is used to ensure convergence and W is the obtained normalization of the adjacency matrix. When column-wise normalization is used, i.e. $W = MD^{-1}$, this converges to ranks proportional the previous ones. Alternatively, we can plug in the symmetric normalization of the adjacency matrix $W = D^{-1/2}MD^{-1/2}$.

Heat Kernels. The node ranks produced by personalized PageRank at each iteration are proportional to the number of random walks through the graph that those nodes. However, it has been argued that shorter walks may be more important than longer ones. This observation has led to the introduction of Heat Kernels (Kloster and Gleich 2014) as an exponential degradation filter of random walk importance $r = e^{-t}\sum_{k=0}^{N}\frac{t^k}{k!}(D^{-1/2}MD^{-1/2})^k q$, which places higher importance on shorter random walks instead of uniformly spreading importance across all walks. This ranking strategy favors local structures at the cost of not spreading ranks too much, according to which distance is considered most important in the parameter t. In our experiments, we try the values $t \in \{1, 3, 5, 7\}$.

Contrary to the original formulation of personalized PageRank, Heat Kernels assume a symmetric propagation of node importance. However, they too can be generalized to allow any kind of adjacency matrix normalization, yielding:

$$r = e^{-t}\sum_{k=0}^{N}\frac{t^k}{k!}W^k q = e^{-t(I-W)}q \tag{11}$$

where W is the normalized adjacency matrix, computed similarly to before. We assume this method to converge when additional summed terms became too small at $\left\|(D^{-1/2}MD^{-1/2})^k q\right\| \leq 10^{-6}$.

Heuristic PageRank Adaptation. We also provide a heuristic adaptation of personalized PageRank that borrows assumptions of heat kernels to place emphasis on short random walks: $r \leftarrow \frac{1}{k}a(W-I)r + (1-a)q$, where k is the current iteration of repeating this formula, W is one of the two normalization of the adjacency matrix and the values of a and convergence criterion are the same as for personalized PageRank.

Augmenting Example Nodes. Finally, there exist several strategies that aim to improve the efficacy of ranking algorithms when the number of example nodes is small. Such strategies work well on few example nodes, but could reduce rank quality by introducing potentially erroneous information when examples suffice to produce high quality ranks. In this work, besides running the above three ranking algorithms with all combinations of described parameters and adjacency matrix normalizations, we also employ three strategies to increase the number of examples. The first of these strategies aims to "inflate" the set of example nodes by adding all their neighbors as examples too (Whang et al. 2016). The idea behind this approach is that, if metadata groups form dense communities, then the immediate neighbors of known members are also likely members. The second strategy aims to "oversample" example nodes by running ranking algorithms once and adding as examples the nodes that are more relevant to the metadata group than at least one of the initial examples (Krasanakis et al. 2019b). The third strategy consists of running ranking algorithms without a search for additional examples.

Measures

The unsupervised measures computed in our experiments are those previously summarized in Table 1. For the proposed local structure measures proposed in this work we employ cosine similarity between potentially linked nodes by additionally L2-normalizing the ranks of each node over metadata groups. To scale down the number of computations needed to calculate these measures, we also use the previously described sampling procedure and evaluate them on $2K$ nodes and equally many negative link nodes for a total of $4M$ potential links. For the types of measures that do not systemically account for multiple communities, their outcome is averaged across all metadata groups.

The efficacy of unsupervised measures is compared with the AUC and Normalized Discounted Cumulative gain (NDCG) of metadata groups. AUC has already been presented in "Local structure evaluation of node ranks" section as a robust non-parametric measure of rank quality. NDCG is also non-parametric statistic that serves a similar purpose. In particular, it derives rank ordinalities $ord_i[j]$ for nodes j (i.e. the highest ranked node is assigned $ord[j] = 1$ and the others increasingly larger integer values) for each metadata group i, assigning to nodes j relevance scores of 1 if they belongs to it and 0 otherwise:

$$NDCG_i = \frac{\sum_{j:j \in eval_i} 1/log_2(ord[j]+1)}{\sum_{c=1}^{|eval_i|} 1/log_2(c+1)} \tag{12}$$

where $eval_i$ denotes the nodes of each metadata group whose labels are withheld for evaluation purposes. NDCG is usually used to evaluate whether a fixed top-k nodes are relevant to the metadata group. However, in this work we are interested in the relevant nodes across the whole network and hence we make this measure span all nodes. This makes it similar to AUC in that values closer to 1 indicate that metadata group members are ranked as more relevant to the group compared to non-group members. Its main difference is that more emphasis is placed on the top discoveries.

As with unsupervised measures that do not systemically account for multiple communities, AUC and NDCG are averaged across metadata groups to yield one value for each ranking algorithm. They both lie in the range $[0, 100\%]$ where larger values indicate node ranks of higher quality.

Measure comparison

A final aspect of our experiments is how to compare unsuprvised with supervised measures, so as to determine the best unsupervised ones. We recognize two qualitative axes that are of practical use: a) yielding the same pairwise comparisons between all pairs of algorithms in the same experiment setting and b) the more specific subtask of selecting similar algorithms of similar quality to the best ones.

In more detail, we begin by exploring the ability of unsupervised measures to mimic supervised ones in comparing the quality of node ranks. They can be considered to succeed in this task if they yield similar ordering of ranking algorithms in the same experiment settings. For example, if we consider pairs of algorithms A and B ran in the same setting and the supervised measure s and unsupervised measure u, the latter can be considered of high quality for comparing any pair of ranking algorithms if on the same network $s(A) < s(B)$ tends to also yield $u(A) < u(B)$ and conversely, i.e. its assessments are correlated to the supervised assessments. We explore whether the above property

holds true for each experiment setting separately, as measure outcome could be influenced by the number of network nodes or example nodes. To summarize the adherence to this property, we calculate the Spearman correlation between the supervised and unsupervised measures, which is a non-parametric metric that compares the ordinality of measure outcomes without being affected by non-linearity.

On the other hand, from a practical standpoint, researchers are likely to deploy procedures similar to ours in listing a large number of promising ranking algorithms and parameters and exploring which ones are best suited to ranking the relatedness of nodes to their metadata groups. In those cases, it is less important to accurately compare all pairs of algorithms and more important to find the algorithms (and hence node ranks) exhibiting close to optimal values. For example, if two algorithms are known to be of low quality, we do not necessarily need to know which of the two is better. Instead, what matters most is for the selected algorithm to be similar to the best one. Recognizing this point leads us to exploring how close unsupervised measures come to finding the algorithms found best by supervised evaluation. To quantify this question, we consider the gaps between the best supervised evaluation and the supervised evaluation for the algorithm suggested as best by unsupervised measures.

Smaller gaps reflect the ability of unsupervised measures to recommend as best the ranking algorithms whose ground truth performance, as obtained by supervised measures, is near-best. For example, let us consider an unsupervised measure u that scores the quality of five algorithms' ranks as $[10\%, 40\%, 60\%, 80\%, 50\%]$ (e.g. 10% is how good the first algorithm is assessed as by that measure), an unsupervised measure v that scores the quality of the same algorithms' ranks as $[50\%, 20\%, 30\%, 20\%, 40\%]$ and a supervised measure g that scores the quality of the same algorithms' ranks as $[50\%, 20\%, 80\%, 70\%, 40\%]$. In this example, u selects the fourth algorithm as the best one. The ground truth quality of that algorithm is calculated by the supervised measure g as 70% and hence lags only 10% behind than the algorithm with the maximum ground truth quality obtained with the same measure, which is 80%. Then, the gap of measure u is calculated as 10%. In this process, it is important to note that, even if measure v performs more accurate pairwsise comparisons between individual algorithms (e.g. it correctly identifies that the first is better than the last one), it selects the first one as the best algorithm, which yields a higher 20% gap.

Results

Comparing any pair of ranking algorithms

We start by comparing the order of ranking algorithms arising from unsupervised measures vs. the ordering provided by AUC and NDCG. As we explain in "Measure comparison" section, when this comparison exhibits strong Spearman correlation, the examined unsupervised measures mimic well the supervised ones that serve as ground truth. In Tables 3 and 4 we show the outcome of this comparison, where higher correlations indicate stronger agreement.[5]

Contrary to the findings of our previous work, LinkAUC is rarely the best unsupervised measure (bolded). Furthermore, unsupervised measures exhibit small correlation with supervised ones in many experiment settings. To make matters worse, which algorithm

[5] The sign of conductance is inverted when measuring its correlation with supervised measures, since otherwise lower conductance corresponds to higher perceived rank quality.

Table 3 Spearman correlation between other measures and AUC

		Structural Measures			Local Structure Measures			
Network	Examples	-Cond/nce	Density	Modularity	LinkCC	LinkAUC	HopAUC	NDCG
Amazon	0.1%	34%	11%	-35%	54%	68%	**79%**	88%
Amazon	1%	16%	-5%	51%	1%	96%	45%	89%
Amazon	10%	-34%	28%	0%	28%	92%	38%	78%
Amazon	25%	-30%	18%	-49%	74%	**84%**	33%	92%
Amazon	50%	-42%	17%	35%	71%	**83%**	58%	92%
DBLP	0.1%	7%	-47%	-50%	26%	**52%**	42%	91%
DBLP	1%	46%	8%	-22%	-85%	28%	**85%**	84%
DBLP	10%	2%	-23%	15%	-55%	-36%	**88%**	32%
DBLP	25%	4%	-28%	5%	-35%	-23%	79%	16%
DBLP	50%	-5%	-25%	12%	-5%	-12%	**43%**	68%
BlockModel	0.1%	12%	42%	21%	-6%	**50%**	49%	80%
BlockModel	1%	-8%	**29%**	23%	-35%	8%	6%	95%
BlockModel	10%	-18%	38%	37%	-40%	40%	**47%**	99%
BlockModel	25%	-63%	**70%**	35%	-77%	-23%	3%	100%
BlockModel	50%	-70%	**72%**	73%	-79%	-22%	0%	-25%
CiteSeer	0.1%	56%	40%	45%	**63%**	39%	**63%**	82%
CiteSeer	1%	19%	41%	47%	**75%**	63%	**75%**	63%
CiteSeer	10%	**70%**	55%	69%	52%	13%	14%	93%
CiteSeer	25%	**42%**	34%	48%	25%	9%	18%	82%
CiteSeer	50%	14%	-7%	22%	41%	35%	**47%**	78%
PubMed	0.1%	48%	-8%	44%	**70%**	-10%	-33%	93%
PubMed	1%	**47%**	27%	46%	37%	-52%	-53%	78%
PubMed	10%	69%	-22%	**78%**	38%	-9%	-14%	81%
PubMed	25%	54%	-22%	**77%**	18%	21%	20%	84%
PubMed	50%	13%	7%	**25%**	-5%	18%	17%	80%

The value unsupervised measure closest to 100% in each setting (bolded) is the one best mimicking AUC for comparing pairs of ranking algorithms

is the best for each setting doesn't appear to follow a consistent pattern but varies both between networks and number of example nodes.

On the other hand, the strongest correlations between local structure and supervised measures tend to be exhibited when only a few (e.g 0.1% or 1% of metadata group size) example nodes are used to calculate ranks. In these cases the best measure is always a local structure one, usually either LinkAUC or HopAUC. Furthermore, these two measures frequently exhibit strong correlation to the supervised evaluation, which far outperforms other measures. Therefore, this type of node rank quality evaluation should be preferred compared to structural measures when the lack of examples leaves no other alternative than unsupervised evaluation of node rank quality.

To understand why unsupervised measures often fail to follow supervised ones, we argue that many ranking algorithms and parameters of those used in our experiments model similar implicit structural assumptions. For example, running the personalized PageRank algorithm with the same adjacency matrix normalization and dampening factors 0.85 and 0.9 diffuses the example nodes' metadata group properties a similar number of hops away from the example nodes. As a result, many node ranking algorithms calculate ranks of similar quality. Then, pairwise comparisons of similar rank qualities can be easily perturbed by systemic noise (e.g. incomplete real-world information) or the random nature of negative sampling[6], which is nevertheless necessary for scalable evaluation. We

[6] All experiments were run with the same random seeds. However, selecting a different number of example nodes also influences the ranks of negative examples so that the outcome of negative sampling remains unpredictable between different experiment settings on the same network.

Table 4 Spearman correlation between other measures and NDCG

		Structural Measures			Local Structure Measures			
Network	Examples	-Cond/nce	Density	Modularity	LinkCC	LinkAUC	HopAUC	AUC
Amazon	0.1%	8%	7%	-1%	78%	34%	**91%**	88%
Amazon	1%	29%	9%	30%	28%	**80%**	67%	89%
Amazon	10%	-42%	22%	-2%	59%	**67%**	-3%	78%
Amazon	25%	-23%	12%	-43%	**77%**	64%	9%	92%
Amazon	50%	-41%	12%	36%	55%	**75%**	54%	92%
DBLP	0.1%	-12%	-48%	-27%	25%	**64%**	55%	91%
DBLP	1%	14%	11%	-12%	9%	39%	**66%**	84%
DBLP	10%	-5%	20%	-57%	-10%	15%	**33%**	32%
DBLP	25%	2%	36%	31%	-11%	-11%	-6%	16%
DBLP	50%	-28%	-19%	17%	-29%	39%	**49%**	68%
BlockModel	0.1%	20%	11%	31%	-9%	20%	**23%**	80%
BlockModel	1%	-8%	25%	24%	-27%	**26%**	22%	95%
BlockModel	10%	-16%	35%	38%	-40%	39%	46%	99%
BlockModel	25%	-64%	**72%**	-78%	-77%	-22%	4%	100%
BlockModel	50%	72%	-52%	20%	62%	**75%**	**75%**	-25%
CiteSeer	0.1%	51%	16%	40%	62%	73%	**89%**	82%
CiteSeer	1%	10%	31%	35%	62%	**67%**	56%	63%
CiteSeer	10%	**64%**	52%	**64%**	54%	27%	26%	93%
CiteSeer	25%	14%	**54%**	31%	50%	-23%	-3%	82%
CiteSeer	50%	-20%	27%	-9%	**79%**	10%	44%	78%
PubMed	0.1%	57%	12%	26%	**60%**	5%	-7%	93%
PubMed	1%	**53%**	16%	**53%**	36%	-11%	-8%	78%
PubMed	10%	60%	-39%	**78%**	33%	31%	26%	81%
PubMed	25%	24%	3%	**52%**	-6%	32%	30%	84%
PubMed	50%	-15%	**32%**	-17%	-48%	-11%	-11%	80%

The value unsupervised measure closest to 100% in each setting (bolded) is the one best mimicking NDCG for comparing pairs of ranking algorithms

theorize that these perturbations of node rank quality assessments affect local structure measures for few example nodes less, because the latter are likely to be well-separated, which makes it easier to identify erroneous diffusion of ranks that occur when random walks starting from one metadata group venture too closely to another.

As a surface-level investigation of these claims, we also present the correlation between supervised measures. These tend to be more strongly correlated, but still differ sometimes. In part, this phenomenon could be blamed on NDCG placing more emphasis on the top node discoveries. However, it also indicates that there could exist multiple views of high rank quality. Hence, even weaker correlations with unsupervised measures can be considered an encouraging -albeit not conclusive- indication of the latter's ability to assess node rank quality.

To continue this investigation at a more fundamental level, we also delve into a detailed view of measure comparisons instead of summarizing their correlation. To this end, in Fig. 10 we present scatterplots between AUC and HopAUC in two experiment settings, where they exhibit 49% and 63% Spearman correlation respectively. At a first glance, the correlation between these two measures is as weak as implied by the Spearman correlation. However, there is a clear trend for their near-top ranking algorithm suggestions to coincide, i.e. there exist algorithms for which both measures exhibit their near-largest value. These findings corroborate our claim that several ranking algorithms exhibit similar node rank quality.

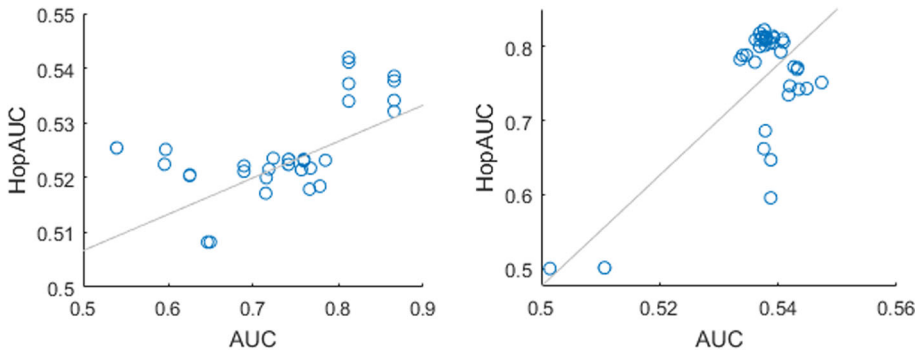

Fig. 10 Scatterplots and least square lines between AUC and HopAUC for the BlockModel with 0.1% example nodes (left) and CiteSeer with 0.1% example nodes (right). Each point corresponds to a different ranking algorithm

Unsupervised selection of near-best ranking algorithms

In the previous subsection we saw that even the best unsupervised measures do not always exhibit strong correlation with supervised evaluation, since they encounter randomness-related difficulties in resolving comparisons between ranking algorithms of similar quality. However, this does not preclude that some of them can help select ranking algorithms of near-top quality, if closeness is not determined by the order arising from pairwise algorithm comparisons but rather by how close calculated node ranks are to the best ones out of all compared ranking algorithms.

We assess this property by continuing with the methodology described in "Measure comparison" section, where we proposed calculating the gap of the best supervised evaluation between with the one arising from the algorithms selected with the help of unsupervised measures. The gaps of AUC and NDCG in all experiment settings are presented in Tables 5 and 6 respectively. Gaps closer to zero indicate that the unsupervised measures help select algorithms that produce node ranks of similar quality to the best algorithm.

To understand which gaps indicate near-optimal node quality, we also provide the ones occurring between supervised measures, i.e. the AUC gap between its highest value and the value of the algorithm with the highest NDCG, as well as the NDCG gap between its highest value and the value of the algorithm with the highest AUC. We consider both types of measures to capture different aspects of node rank quality and hence their maximal difference indicates which value gaps should be considered of high quality. This exploration indicates that gaps up to 5% for AUC and 9% for NDCG can be considered to indicate high node rank quality. We heuristically set thresholds at 1.5 times those values to recognize algorithms that exhibit near-top rank quality.

Overall, HopAUC is the measure that finds the most high quality algorithms (20 for AUC and NDCG) and the most near-top node ranking algorithms (23 for AUC and 20 for NDCG). Modularity and LinkAUC follow as the second-best for AUC and NDCG respectively, even if in some experiments they do not exhibit strong correlation with the supervised measures for pairwise algorithm comparisons. Although conductance, density and LinkCC also exhibit small evaluation gaps at times, they do so with a much lower frequency.

Table 5 Gap between the best AUC value and the AUC of the algorithm corresponding to the best measure value

		Structural Measures			Local Structure Measures			
Network	Examples	Cond/nce	Density	Modularity	LinkCC	LinkAUC	HopAUC	NDCG
Amazon	0.1%	24%	23%	23%	**0%**	**0%**	**0%**	0%
Amazon	1%	20%	6%	12%	6%	**0%**	**0%**	3%
Amazon	10%	9%	4%	6%	7%	1%	2%	0%
Amazon	25%	5%	4%	27%	**0%**	2%	2%	1%
Amazon	50%	17%	3%	3%	3%	2%	**2%**	1%
DBLP	0.1%	11%	10%	**6%**	9%	**6%**	**6%**	0%
DBLP	1%	4%	11%	1%	15%	**0%**	**0%**	0%
DBLP	10%	1%	5%	**1%**	2%	**1%**	**1%**	5%
DBLP	25%	0%	2%	**0%**	2%	**0%**	1%	2%
DBLP	50%	1%	2%	**1%**	**1%**	**1%**	**1%**	0%
BlockModel	0.1%	33%	12%	33%	33%	**5%**	**5%**	0%
BlockModel	1%	29%	7%	**0%**	36%	6%	6%	0%
BlockModel	10%	33%	8%	**4%**	40%	13%	5%	0%
BlockModel	25%	35%	**5%**	9%	43%	10%	10%	0%
BlockModel	50%	36%	4%	**1%**	42%	9%	**1%**	1%
CiteSeer	0.1%	4%	1%	**0%**	1%	1%	1%	1%
CiteSeer	1%	1%	**0%**	1%	4%	3%	4%	2%
CiteSeer	10%	2%	2%	2%	**1%**	2%	2%	0%
CiteSeer	25%	3%	**2%**	3%	3%	3%	3%	1%
CiteSeer	50%	**3%**	4%	**3%**	**3%**	**3%**	**3%**	1%
PubMed	0.1%	9%	13%	**2%**	22%	19%	18%	0%
PubMed	1%	**7%**	9%	**7%**	21%	12%	**7%**	1%
PubMed	10%	**4%**	6%	**4%**	**4%**	9%	9%	0%
PubMed	25%	**4%**	8%	**4%**	27%	**4%**	**4%**	0%
PubMed	50%	5%	11%	5%	27%	4%	**3%**	0%
Number of gaps $\leq$ 5%		11	13	18	12	17	20	25
Number of gaps $\leq$ 7%		12	16	22	14	19	23	25

Gaps closer to zero mean that the AUC found when selecting the algorithm optimizing the respective measure is close to the max AUC between all algorithms. The smallest gap among unsupervised measures in each experiment setting is bolded

We finally assert that the correctness of pairwise comparisons is not indicative of a measure's ability to select the best algorithm. For example, in the BlockModel experiments density often exhibits the strongest correlation with supervised measures for pairwise algorithm comparison, but is outperformed in all but one experiment settings by HopAUC for finding small gaps.

Discussion

Which measure to use

In our experiments, HopAUC is the measure that leads to selecting high-quality node ranking algorithms the most times. This result holds true across all three network domains involved in our experiments; in the Amazon network, whose edges are formed due to similar behavior of their endpoints (i.e. being co-purchased), in the citation networks, whose edges are formed incrementally as more papers are introduced in the literature and cite previous ones, and in the synthetic network whose edges are formed through a stochastic block model. In most cases, LinkAUC performs similarly well and can hence serve as a substitute that is faster to calculate.

On the other hand, both of these measures sometimes yield an erroneous selection of the best-performing ranking algorithm in the BlockModel and PubMed networks.

Table 6 Gap between the best NDCG value and the NDCG of the algorithm corresponding to the best measure value

		Structural Measures			Local Structure Measures			
Network	Examples	Cond/nce	Density	Modularity	LinkCC	LinkAUC	HopAUC	AUC
Amazon	0.1%	24%	23%	13%	**0%**	**0%**	**0%**	0%
Amazon	1%	5%	4%	5%	4%	1%	**1%**	1%
Amazon	10%	9%	4%	7%	**2%**	**2%**	6%	0%
Amazon	25%	10%	14%	19%	**2%**	6%	6%	1%
Amazon	50%	35%	13%	17%	12%	**4%**	**4%**	2%
DBLP	0.1%	8%	5%	**3%**	7%	**3%**	4%	0%
DBLP	1%	4%	5%	3%	7%	15%	**0%**	0%
DBLP	10%	3%	**0%**	9%	5%	**0%**	**0%**	4%
DBLP	25%	7%	**3%**	9%	6%	10%	15%	9%
DBLP	50%	9%	20%	11%	**5%**	7%	7%	21%
BlockModel	0.1%	32%	18%	32%	8%	**8%**	8%	0%
BlockModel	1%	32%	14%	**0%**	39%	9%	10%	0%
BlockModel	10%	48%	20%	**11%**	51%	31%	16%	0%
BlockModel	25%	45%	17%	24%	27%	24%	24%	0%
BlockModel	50%	27%	38%	**0%**	28%	14%	2%	12%
CiteSeer	0.1%	10%	3%	3%	**1%**	2%	2%	3%
CiteSeer	1%	3%	4%	3%	4%	**2%**	4%	3%
CiteSeer	10%	10%	5%	10%	**2%**	6%	6%	6%
CiteSeer	25%	12%	5%	12%	**3%**	8%	8%	6%
CiteSeer	50%	7%	10%	7%	**2%**	3%	3%	1%
PubMed	0.1%	6%	13%	**2%**	12%	19%	18%	0%
PubMed	1%	**6%**	9%	7%	6%	12%	7%	1%
PubMed	10%	8%	6%	**4%**	19%	9%	9%	0%
PubMed	25%	12%	8%	4%	29%	**4%**	**4%**	0%
PubMed	50%	7%	11%	5%	22%	4%	**3%**	0%
Number of gaps $\leq$ 9%		13	13	17	16	19	20	23
Number of gaps $\leq$ 13%		18	17	22	18	21	20	24

Gaps closer to zero mean that the NDCG found when selecting the algorithm optimizing the respective measure is close to the max NDCG between all algorithms. The smallest gap among unsupervised measures in each experiment setting is bolded

This can be attributed to edges of these networks forming for reasons other than metadata group memberships. For example, in the BlockModel network, the nodes of some metadata group pairs (i.e. the group pairs $(1, 2)$ and $(2, 5)$ in Fig. 9) are more closely connected to each other than internally. This creates an alternate view in which those pairs of metadada groups can be merged into larger ones but which cannot be explicitly discouraged by the -otherwise random- generation of edges.

Overall, we propose that researchers should start with an empirical investigation of the network's domain that explores whether the metadata groups characterized by example nodes influence or are influenced by the formation of network edges. If this dynamic property is known to be true, and the few example nodes aren't enough for a supervised evaluation, we encourage usage of either LinkAUC or HopAUC for selecting the best ranking algorithms and parameters across potential candidates. In fact, these two measures are at least as strong as the network's homophily (see "Dynamics of local structures" section) and are otherwise not affected by the prevalence of non-local structures fv that often occur in networks of larger diameters. HopAUC can find near-best ranking algorithms when more example nodes are provided too. Nonetheless, the results of pairwise algorithm comparison suggest that local structure measures are most suited to

differentiating between high-quality and low-quality ranking algorithms for few example nodes for each metadata group.

When the network's domain exhibits specific dynamics that drive the formation of edges or organization of nodes into communities, our experiments suggest that using a measure that explicitly captures those could achieve equally good or better results to HopAUC or LinkAUC. For example, modularity does not characterize well the quality of node ranks in the Amazon network, but can be considered the best measure if we focus only on citation networks (including PubMed), potentially due to implicitly modeling that new members of this domain's metadata groups favor links with previous members rather than with random nodes. However, blind use of another measure besides LinkAUC or HopAUC has a higher chance of misidentifying low-quality ranking algorithms as high-quality ones, as can be seen by the frequent inability of conductance and density to identify the best algorithms.

Using unsupervised measures of node rank quality

In practice, our investigation suggests that the best measures of node rank quality for each experiment setting (e.g. HopAUC for most settings) should be used to select the best parameters and algorithms among a predefined set of many candidates. For example, if one were to investigate ranking algorithms that would be used to recommend nodes either to provide more example nodes (Li et al. 2018; Kipf and Welling 2016) or to provide more metadata group members (Klicpera et al. 2018), selecting the algorithms and hyper-parameters that best suit the available relational data could be done with an unsupervised measure. It must be stressed that, due to the uncertain efficacy of unsupervised measures in pairwise ranking algorithm comparisons, many algorithms need be compared to identify the best ones. For example, if a set of ranking algorithms yielded HopAUC of [20%, 38%, 45%, 60%, 70%], the last two ones are likely to yield node ranks of high quality. However, a comparison only involving the third and the fifth algorithm could be misleading, as they don't differ that much.

This setup effectively uses the selected unsupervised measure to guide the more sophisticated learning algorithms. Then, the question arises of whether we could directly set the unsupervised measure of choice as the optimization objective of such algorithms, similarly to approaches that represent nodes in low-dimensional spaces (Menon and Elkan 2011; Tang et al. 2015; Wang et al. 2016; Yang et al. 2017) or as already done by works that greedily expand local structural communities by optimizing unsupervised measures - usually modularity (Kim and Tan 2010; Tabrizi et al. 2013). Besides the computational cost needed in finding gradients over measures such as HopAUC, which involve sorting operations, our experiments show that neither structural nor local structure measures can produce a high quality of pairwise comparison of node ranks. Hence, unsupervised measures do not exhibit a convexity that would allow greedy or stochastic tuning towards an optimal value; such processes can even cause to steer away from a high-quality solution. For example, this would happen if we tried to directly optimize HopAUC in PubMed with 1% examples, where it negatively correlates with supervised measures for pairwise comparisons. On the other hand, if we consider a large space of potential solutions in this example, we obtain a ranking algorithm of $\leq 7\%$ gap. In part, the lack of convexity can be attributed to constraining the space of node representations to very few dimensions (i.e.

equal to the number of metadata groups) whose explicit understanding cannot be altered to better match the latent attributes driving node behavior.

Given these concerns, our suggested usage of unsupervised measures of node rank quality is not as objectives of ranking algorithms but as coarse tuning mechanisms that help select which algorithms and hyper-parameters best match the structure of each network for the ranked metadata groups. For example, despite the success of the aforementioned greedy community expansion in optimizing unsupervised measures, similar processes should not be blindly applied for ranking the relevance of network nodes to metadata groups by greedily optimizing the unsupervised measures examined in this work. Instead, a number of potential solutions need be procured through other means and the best one selected with the measure that matches the network's dynamics driving the formation of edges.

Dynamics of local structures

We previously proposed that the best unsupervised measures for selecting high-quality node ranking algorithms are the ones that best match network dynamics, if these are known. Structural measures of community quality, i.e. conductance, density and modularity, directly capture their favored type of communities and are hence easy to identify. On the other hand, the evaluation of local structures relies on correctly assessing node rank quality based on underlying dynamics, which we outline here.

To begin with, the theoretical probing of "Approach motivation and overview" section reveals that LinkAUC simultaneously assesses whether the similarity of node rank distributions can reconstruct the network's edges given that nodes of similar attributes tend to link to each other. This statement's precondition translates to the notion of homophily (McPherson et al. 2001; Berry et al. 2020), which assumes that linked nodes influence each other towards obtaining similar attributes. Then, the assessment of LinkAUC can be of high quality only for networks of high homophily. An important distinction to make is that this does not necessarily pertain to any heuristic measure of *observed* homophily (Berry et al. 2020), such as the one reported in Table 2. In particular, the latter estimates only a lower bound of actual homophily that does not account for either combinations of metadata attributes not captured by linking only nodes of the same groups (e.g. if both political and religious views were need to coincide for social media users to link to each other) or missing metadata groups, as happens for our experiments on the DBLP network. In practical applications of our work, where the absence of node labels may prevent any type of homophily-related computation in the first place, it is preferable to identify homophilous networks by understanding whether the metadata groups of the example nodes can influence or be influenced by the formation of network edges through real-world processes.

Secondly, HopAUC extends the evaluation of LinkAUC to also account for nodes that reside up to two hops away in the network's structure. In addition to homophily, this definition also entertains the notion of nodes matching the attributes of their neighbors-of-neighbors, a property often referred to as structural equivalence between nodes of similar attributes (Friedkin 1984; Burt 1987; Kuwashima 2016; Shi et al. 2019). The most widely accepted interpretation of structural equivalence is that nodes of similar attributes are attracted towards neighbors of similar attributes, even if they are not linked. This

differs from homophily in that the immediate neighbors are not necessarily similarly-attributed. A local interpretation of this phenomenon would be that nodes of similar attributes often end up linking to the same neighbors (Simões et al. 2019). As a result, HopAUC can produce an accurate assessment of node rank quality when network edges are correlated to a combination of homophily and structural equivalence; either of these dynamics can justify the application of this measure, thus extending the types of networks it can be applied on.

Lastly, the triadic closure that characterizes LinkCC assesses whether similarities between nodes can replicate both their edges and the edges between each node's neighbors. For accurate replications to indicate high rank quality, network nodes should exhibit both homophily, so that similarly-attributed neighbors are linked, and structural equivalence, so that linked nodes (which exhibit similar ranks due on homophily) are also linked through common neighbors. As a result, LinkCC requires these two different dynamics to hold true at the same time, which inhibits its adoption in many networks and can explain its limited efficacy in our experiments.

Threats to validity

As we mention above, our recommended general-purpose LinkAUC and HopAUC measures heavily rely on the notion of homophily between network nodes, in the sense that nodes of similar relatedness to metadata groups are often linked to each other. As a result, the burden of identifying that this is one of a network's dynamics falls on the one applying our methodology. In this regard, a vague consensus exists in the complex network community that homophily characterizes many systems, although it has been corroborated mostly on social networks (McPherson et al. 2001; Aiello et al. 2012; Dehghani et al. 2016; Huber and Malhotra 2017). The usage of HopAUC can also be justified through structural equivalence between nodes of similar attributes, but the extend to which this dynamic permeates all kinds of real-world networks has, to our knowledge, not been studied yet. Hence, there exist concerns over the efficacy of using the strategies proposed in this work on other types of domains in which local structures cannot be systemically corroborated. Of course, these concerns are no greater than for the assumptions of other unsupervised measures.

Another potential risk lies in our methodology for selecting the most promising unsupervised evaluation measure that finds a near-best link ranking algorithm. In particular, there could exist systemic properties of the ranking algorithms we deploy that favor the usage of certain measures. For example, in the case of HopAUC, disseminating ranks to neighbors-of-neighbors could be the most important challenge the node ranking algorithms we examine (but not all ranking algorithms) need overcome. Although we tried to mitigate this phenomenon by employing many ranking algorithms and variations, ultimately these all work by defusing the metadata group membership of example nodes in the network.

Finally, we stress that the suggestion of using either HopAUC or the similarly-performing LinkAUC to find the best ranking algorithm relies on the assumption that our experiments cover an adequately wide range of potentially high quality algorithms.

Conclusions and future work

In this work we explored scalable unsupervised procedures that evaluate node ranks of multiple metadata groups based on how well they predict the network's local structures. We explained the intuitive motivation behind these approaches and experimentally showed that they are often better than structural community quality measures for comparing the quality of two possible ranking algorithms. We also found that the measure evaluating structural proximity up to two hops away, which we call HopAUC, is the most promising for selecting node ranks of the highest quality in new networks when possible candidates are obtained by many different algorithms.

Future research can move in the direction of providing a unified framework between the unsupervised measures presented in this work, for example by combining their assessments. Furthermore, semi-supervised network mining algorithms that involve unsupervised extraction of node ranks could be augmented with unsupervised measures of rank quality that select a different node ranking algorithm for each network.

Abbreviations

AUC: Area Under Curve of the receiver operating characteristics; NDCG: Normalized Discounted Cumulative Gain; aCC: Heuristic adaptation of the Clustering Coefficient; HopAUC: AUC of using node ranks to predict nodes up to two hops away; LinkAUC: AUC of using node ranks to predict network edges; LinkCC: Fraction of the network's aCC discovered by ranks

Acknowledgements

Not applicable.

Authors' contributions

All authors developed the theoretical concepts and designed the experiments discussed in the paper. EK wrote the code and performed the data analysis. EK and SP wrote the manuscript. SP and YK supervised the work. All authors read and approved the final manuscript.

Funding

This work was partially funded by the European Commission under contract numbers H2020-761634 FuturePulse and H2020-825585 HELIOS.

Availability of data and materials

The ranking algorithms and evaluation measures developed in this research can be found in the pygrank repository (Python Graph Ranking (pygrank) library 2019). The preprocessing of the public datasets used and analysed during the current study to convert them into a common format is available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Received: 28 February 2020 Accepted: 17 July 2020
Published online: 06 August 2020

References

Abbe E, Bandeira AS, Hall G (2016) Exact recovery in the stochastic block model. IEEE Trans Inf Theory 62(1):471–487
Aiello LM, Barrat A, Schifanella R, Cattuto C, Markines B, Menczer F (2012) Friendship prediction and homophily in social media. ACM Trans Web (TWEB) 6(2):1–33
Amazon product co-purchasing network metadata (2007). https://snap.stanford.edu/data/amazon-meta.html. Accessed 28 Feb 2020
Andersen R, Chung F, Lang K (2006) Local graph partitioning using pagerank vectors. In: 2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06). IEEE, New York. pp 475–486
Andersen R, Chung F, Lang K (2008) Local partitioning for directed graphs using pagerank. Internet Math 5(1-2):3–22
Avrachenkov K, Kadavankandy A, Litvak N (2018) Mean field analysis of personalized pagerank with implications for local graph clustering. J Stat Phys 173(3-4):895–916
Berry G, Sirianni A, Weber I, An J, Macy M (2020) Going beyond accuracy: estimating homophily insocial networks using predictions. arXiv preprint arXiv:2001.11171
Burt RS (1987) Social contagion and innovation: Cohesion versus structural equivalence. Am J Sociol 92(6):1287–1335
Chalupa D (2017) A memetic algorithm for the minimum conductance graph partitioning problem. arXiv preprint arXiv:1704.02854
CiteSeer network (2003). CiteSeer for Document Classification from https://linqs.soe.ucsc.edu/data. Accessed 28 Feb 2020
Dash NS (2018) Context and contextual word meaning. SKASE J Theor Linguist 2:21–31
DBLP Citation network (2011). DBLP-Citation-network V4 from https://aminer.org/citation. Accessed 28 Feb 2020

De Domenico M, Solé-Ribalta A, Omodei E, Gómez S, Arenas A (2015) Ranking in interconnected multilayer networks reveals versatile nodes. Nat Commun 6:6868
Dehghani M, Johnson K, Hoover J, Sagi E, Garten J, Parmar NJ, Vaisey S, Iliev R, Graham J (2016) Purity homophily in social networks. J Exp Psychol Gen 145(3):366
Duan L, Ma S, Aggarwal C, Ma T, Huai J (2017) An ensemble approach to link prediction. IEEE Trans Knowl Data Eng 29(11):2402–2416
Fortunato S, Hric D (2016) Community detection in networks: A user guide. Phys Rep 659:1–44
Friedkin NE (1984) Structural cohesion and equivalence explanations of social homogeneity. Sociol Methods Res 12(3):235–261
Goldberg Y, Levy O (2014) word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. arXiv preprint arXiv:1402.3722
Görke R, Kappes A, Wagner D (2015) Experiments on density-constrained graph clustering. J Exp Algorithmics (JEA) 19:3–3
Hanley JA, McNeil BJ (1982) The meaning and use of the area under a receiver operating characteristic (roc) curve. Radiology 143(1):29–36
Hoff P (2008) Modeling homophily and stochastic equivalence in symmetric relational data. In: Advances in Neural Information Processing Systems. MIT Press, Cambridge. pp 657–664
Holland PW, Laskey KB, Leinhardt S (1983) Stochastic blockmodels: First steps. Soc Netw 5(2):109–137
Hric D, Darst RK, Fortunato S (2014) Community detection in networks: Structural communities versus ground truth. Phys Rev E 90(6):062805
Hric D, Peixoto TP, Fortunato S (2016) Network structure, metadata, and the prediction of missing nodes and annotations. Phys Rev X 6(3):031038
Huber GA, Malhotra N (2017) Political homophily in social relationships: Evidence from online dating behavior. J Polit 79(1):269–283
Hsu C-C, Lai Y-A, Chen W-H, Feng M-H, Lin S-D (2017) Unsupervised ranking using graph structures and node attributes. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. ACM, New York. pp 771–779
Isinkaye F, Folajimi Y, Ojokoh B (2015) Recommendation systems: Principles, methods and evaluation. Egypt Inf J 16(3):261–273
Jeub LG, Balachandran P, Porter MA, Mucha PJ, Mahoney MW (2015) Think locally, act locally: Detection of small, medium-sized, and large communities in large networks. Phys Rev E 91(1):012821
Jin EM, Girvan M, Newman ME (2001) Structure of growing social networks. Phys Rev E 64(4):046132
Keith Borland J (1950) The fallacy of the square root sampling rule. J Am Pharm Assoc 39(7):373–377
Kim J, Tan K (2010) Discover protein complexes in protein-protein interaction networks using parametric local modularity. BMC Bioinformatics 11(1):521
Kipf TN, Welling M (2016) Semi-supervised classification with graph convolutional networks. In: 5th International Conference on Learning Representations (ICLR 2017), Toulon. arXiv preprint arXiv:1609.02907
Klicpera J, Bojchevski A, Günnemann S (2018) Predict then propagate: Graph neural networks meet personalized pagerank, New Orleans. arXiv preprint arXiv:1810.05997
Kloster K, Gleich DF (2014) Heat kernel based community detection. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York. pp 1386–1395
Kowalik Ł (2006) Approximation scheme for lowest outdegree orientation and graph density measures. In: International Symposium on Algorithms and Computation. Springer, Berlin. pp 557–566
Koren Y, Bell R (2015) Advances in collaborative filtering. In: Recommender Systems Handbook. Springer, Boston. pp 77–118
Krasanakis E, Papadopoulos S, Kompatsiaris Y (2019a) LinkAUC: Unsupervised evaluation of multiple network node ranks using link prediction. In: International Conference on Complex Networks and Their Applications, Vol. 1. Springer, Cham. pp 3–14
Krasanakis E, Schinas E, Papadopoulos S, Kompatsiaris Y, Symeonidis A (2019b) Boosted Seed Oversampling. Inf Process Manag 57(2):102053. Elsevier, Amsterdam
Kuwashima Y (2016) Structural equivalence and cohesion can explain bandwagon and snob effect. Ann Bus Adm Sci 15(1):1–14
Lancichinetti A, Fortunato S, Kertész J (2009) Detecting the overlapping and hierarchical community structure in complex networks. New J Phys 11(3):033015
Leskovec J, Adamic LA, Huberman BA (2007) The dynamics of viral marketing. ACM Trans Web (TWEB) 1(1):5
Leskovec J, Lang KJ, Dasgupta A, Mahoney MW (2009) Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. Internet Math 6(1):29–123
Leskovec J, Lang KJ, Mahoney M (2010) Empirical comparison of algorithms for network community detection. In: Proceedings of the 19th International Conference on World Wide Web. ACM, New York. pp 631–640
Levy O, Goldberg Y (2014) Neural word embedding as implicit matrix factorization. In: Advances in Neural Information Processing Systems. MIT Press, Cambridge. pp 2177–2185
Li Q, Han Z, Wu X-M (2018) Deeper insights into graph convolutional networks for semi-supervised learning. In: Thirty-Second AAAI Conference on Artificial Intelligence, Palo Alto
Liben-Nowell D, Kleinberg J (2007) J Am Soc Inf Sci Technol 58(7):1019–1031
Lofgren P, Banerjee S, Goel A (2016) Personalized pagerank estimation and search: A bidirectional approach. In: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining. ACM, New York. pp 163–172
Lü L, Zhou T (2011) Link prediction in complex networks: A survey. Phys A Stat Mech Appl 390(6):1150–1170
Martínez V, Berzal F, Cubero J-C (2017) A survey of link prediction in complex networks. ACM Comput Surv (CSUR) 49(4):69
Mason SJ, Graham NE (2002) Areas beneath the relative operating characteristics (roc) and relative operating levels (rol) curves: Statistical significance and interpretation. Q J R Meteorol Soc 128(584):2145–2166
McPherson M, Smith-Lovin L, Cook JM (2001) Birds of a feather: Homophily in social networks. Annu Rev Sociol 27(1):415–444
Menon AK, Elkan C (2011) Link prediction via matrix factorization. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Berlin. pp 437–452

Namata G, London B, Getoor L, Huang B, EDU U (2012) Query-driven active surveying for collective classification. In: 10th International Workshop on Mining and Learning with Graphs, vol. 8
Newman ME (2006) Modularity and community structure in networks. Proc Natl Acad Sci 103(23):8577–8582
Opsahl T, Panzarasa P (2009) Clustering in weighted networks. Soc Netw 31(2):155–163
Ortega A, Frossard P, Kovačević J, Moura JM, Vandergheynst P (2018) Graph signal processing: Overview, challenges, and applications. Proc IEEE 106(5):808–828
Papadopoulos S, Kompatsiaris Y, Vakali A, Spyridonos P (2012) Community detection in social media. Data Min Knowl Disc 24(3):515–554
Peel L, Larremore DB, Clauset A (2017) The ground truth about metadata and community detection in networks. Sci Adv 3(5):1602548
Perer A, Shneiderman B (2006) Balancing systematic and flexible exploration of social networks. IEEE Trans Vis Comput Graph 12(5):693–700
PubMed network (2012). PubMed Diabetes from https://linqs.soe.ucsc.edu/data. Accessed 28 Feb 2020
Python Graph Ranking (pygrank) library (2019). https://github.com/MKLab-ITI/pygrank. Accessed 28 Feb 2020
Rohe K, Chatterjee S, Yu B, et al. (2011) Spectral clustering and the high-dimensional stochastic blockmodel. Ann Stat 39(4):1878–1915
Schaeffer SE (2007) Graph clustering. Comput Scie Rev 1(1):27–64
Sen P, Namata G, Bilgic M, Getoor L, Galligher B, Eliassi-Rad T (2008) Collective classification in network data. AI Mag 29(3):93–93
Shani G, Gunawardana A (2011) Evaluating recommendation systems. In: Recommender Systems Handbook. Springer, Berlin. pp 257–297
Shi B, Zhou C, Qiu H, Xu X, Liu J (2019) Unifying structural proximity and equivalence for network embedding. IEEE Access 7:106124–106138
Simões JE, Figueiredo DR, Barbosa VC (2019) Local symmetry in random graphs, IEEE Transactions on Network Science and Engineering. IEEE, New York. https://doi.org/10.1109/TNSE.2019.2957610
Stanford Network Analysis Project (SNAP) datasets (2009). https://snap.stanford.edu/data/. Accessed 28 Feb 2020
Tabrizi SA, Shakery A, Asadpour M, Abbasi M, Tavallaie MA (2013) Personalized pagerank clustering: A graph clustering algorithm based on random walks. Phys A Stat Mech Appl 392(22):5772–5785
Tan X (2017) A new extrapolation method for pagerank computations. J Comput Appl Math 313:383–392
Tang J, Qu M, Wang M, Zhang M, Yan J, Mei Q (2015) Line: Large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web. ACM, New York. pp 1067–1077
Tang J, Zhang J, Yao L, Li J, Zhang L, Su Z (2008) Arnetminer: extraction and mining of academic social networks. In: Proceedings of the 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. ACM. pp 990–998
Wang D, Cui P, Zhu W (2016) Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York. pp 1225–1234
Wang Y, Wang L, Li Y, He D, Chen W, Liu T-Y (2013) A theoretical analysis of ndcg ranking measures. In: Proceedings of the 26th Annual Conference on Learning Theory (COLT 2013), vol. 8. PMLR, Paris. p 6
Whang JJ, Gleich DF, Dhillon IS (2016) Overlapping community detection using neighborhood-inflated seed expansion. IEEE Trans Knowl Data Eng 28(5):1272–1284
Wu X-M, Li Z, So AM, Wright J, Chang S-F (2012) Learning with partially absorbing random walks. In: Advances in Neural Information Processing Systems. MIT Press, Cambridge. pp 3077–3085
Wu Z, Lin Y, Wang J, Gregory S (2016) Link prediction with node clustering coefficient. Phys A Stat Mech Appl 452:1–8
Xie J, Kelley S, Szymanski BK (2013) Overlapping community detection in networks: The state-of-the-art and comparative study. ACM Comput Surv (CSUR) 45(4):43
Yang J, Leskovec J (2015) Defining and evaluating network communities based on ground-truth. Knowl Inf Syst 42(1):181–213. ACM, New York
Yang C, Sun M, Liu Z, Tu C (2017) Fast network embedding enhancement via high order proximity approximation. In: IJCAI. pp 3894–3900

Publisher's Note