Uncovering the Nucleus of Social Networks

Braulio Dumba, Zhi-Li Zhang
University of Minnesota, Twin Cities, MN, USA
braulio,zhzhang@cs.umn.edu

ABSTRACT

Many social network studies have focused on identifying communities through clustering or partitioning a large social network into smaller parts. While community structure is important in social network analysis, relatively little attention has been paid to the problem of "core structure" analysis in many social networks. Intuitively, one may expect that many social networks possess some sort of a "core" which holds various parts of the network (or constituent "communities") together. We believe that it is just as important to uncover and extract the "core" structure – referred to as the "nucleus" in this paper – of a social network as to identify its community structure. In this paper, we propose a scalable and effective procedure to uncover the "nucleus" of social networks by building upon and generalizing ideas from the existing k-shell decomposition approach. We employ our approach to uncover the nucleus in several example communication, collaboration, interaction, location-based and online social networks. Our methodology is very scalable and can also be applied to massive networks (hundreds million nodes and billion edges).

CCS CONCEPTS

• **Information systems** → **Social networks**; • **Human-centered computing** → *Social network analysis*; • **Theory of computation** → Shortest paths;

KEYWORDS

Social Network; K-Shell Decomposition; Network Core

ACM Reference Format:
Braulio Dumba, Zhi-Li Zhang. 2018. Uncovering the Nucleus of Social Networks. In *WebSci '18: 10th ACM Conference on Web Science, May 27–30, 2018, Amsterdam, Netherlands.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3201064.3201075

1 INTRODUCTION

Networks are often abstractly modelled as a graph where vertices represent entities and edges capture the relations (e.g., connections) or interactions between them. In the context of (online) social networks, community identification has received a lot of attention. A community is often considered to be a subset of vertices that are densely connected internally but sparsely connected to the rest of the network [9, 30, 35–37]. The majority of studies on identifying communities structures in social networks have relied on clustering techniques, namely, by partitioning the underlying network/social graph into *disjoint* (sometimes *overlapping*) communities. For example, Newman proposes a measure of betweenness – modularity [36, 37] – for identifying disjoint communities in a social network. Andersen et al [9] design a local graph partitioning algorithm to indentify community structures. This algorithm is based on personalized PageRank vectors. Ahn et al [6] introduce a novel perspective for discovering hierarchical community structures by categorizing links only. To obtain an optimal partition and to find communities at multiple levels, an information-theoretic framework is proposed by the authors in [38, 40]. Several studies use link and content information for uncovering meaningful communities in networks [22, 50].

Although existing studies of community structure have been very successful, most have not considered the existence of "core structure" in many networks. Intuitively, one expects that many social networks possess some sort of "core" as part of their meso-scale structure, which holds various parts of the network (or constituent "communities") together. We believe that it is just as important to uncover and extract the "core" structure – referred to as the "nucleus" – of a social network as identify its community structure [39, 49]: unlike "ordinary" constituent communities, the "core" structure plays a crucial role in the formation and evolution of a social network, to which other (constituent) "communities" are attached. Chung and Lu [18] show that power-law random graphs almost surely contain a core "subgraph" when the exponent β in the power-law degree distribution is such that $\beta \in (2, 3)$. This theoretical result suggests that many real-world social networks likely posess some sort of cohesive core structure.

One of the most popular notion of network core is given by the *k-shell decomposition* method [15]. This classical graph decomposition technique decomposes a network into hierarchically ordered layers from the periphery to the core. This method has also be extended to weighted graphs [24, 48] and dynamic networks [32]. The k-shell decomposition method has often been used as a visualization tool for studying the core structure of massive complex networks such as the Internet [15]. In addition, it has been used to identify influential spreaders in a network [23, 28].

When applying the standard k-shell decomposition to uncover the core of several example social networks (see § 2), we find that the resulting "innermost" structure is unlikely to represent the "core" of these networks. For example, this "innermost" structure may contain the maximum clique of a network but which lies rather at its periphery, or it is simply a single vertex in a dense graph. This appears to the effect of the (iterative) degree-based pruning process of k-shell decomposition, where despite at some point we reach the vicinity of the core, the k-shell decomposition continues further, which then destroys the "core" structure of the network (see § 3

WebSci '18, May 27–30, 2018, Amsterdam, Netherlands

ACM ISBN 978-1-4503-5563-6/18/05…$15.00
https://doi.org/10.1145/3201064.3201075

for more illustration). This raises the following important question: *When should we stop the k-shell decomposition pruning process in order to preserve the core graph G_C of a network?*

In an attempt to address this question, we develop an effective procedure to uncover the *nucleus* structure of a social network by building upon and generalizing ideas from the existing k-shell decomposition [15] approach, as follows. Firstly, we propose a new metric, the *dependence value*, that measures the location importance of a node in a network. Intuitively, the dependence of node v captures the number of nodes recursively dependent of v that have been removed in earlier steps of the k-shell decomposition method. Secondly, we derive a new measure called *nucleon-index* (NI) that captures the extend to which a subgraph is a densely intra-connected and topological central core. This index can be used with a wide variety of functions to transition between core and peripheral nodes (e.g., dependence value, closeness [41] and betweenness [41] centralities, etc). Using these metrics, we therefore modify the standard k-shell decomposition method to stop the process earlier, in order to extract a meaningful "core" for social networks (see § 4). For a Facebook [4, 29] friendship network composed of 63,731 nodes and 817,035 edges, this process yields a dense "core" subgraph G_C with approximately 285 nodes and 9,616 edges. Given a dense core subgraph G_C, we investigate the importance of this substructure for the network by analysing the following metrics (see § 5): i) the distance between a node v to the core subgraph G_C; ii) the ratio of the distance between nodes u and v to their respective distance to G_C and iii) lastly, the impact of removing G_C in the structure of the network G ($G_C \subset G$).

We discuss implications and related work in § 6 and § 7. Section 8 concludes the paper. We summarize the major contributions of our paper as follows:

- We propose two *new* metrics: i) the *dependence value*, that measures the location importance of a node in the network; ii) the *nucleon-index* (NI) that captures the extend to which a subgraph is a densely intra-connected and topological central core . Using these metrics, we therefore modify the standard k-shell decomposition method to stop the process earlier, in order to extract a meaningful "core" for social networks.
- We apply our approach to uncover the core structure in example communication, collaboration, interaction, location-based and online social networks. Our methodology is very scalable and can also be applied to uncover the core structure of massive networks (hundreds million nodes and billion edges).

2 DATASETS

This section presents a summary of the datasets that we use for our analysis:

Autonomous systems graph: This dataset is an undirected graph of the AS peering information inferred from Oregon route-views between March 31 and May 26, 2001 [2], and its main features are summarized on Table 1.

Social networks graphs: This dataset is a collection of 9 undirected graphs of communication, collaboration, interaction, location-based and online social networks [1–5, 11, 25, 29, 46](see Table 1 for a summary of the main features):

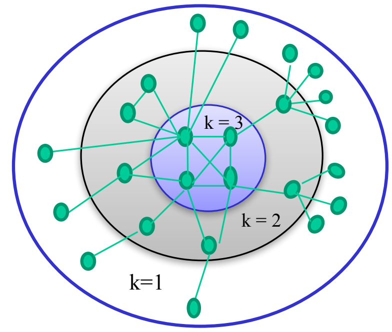
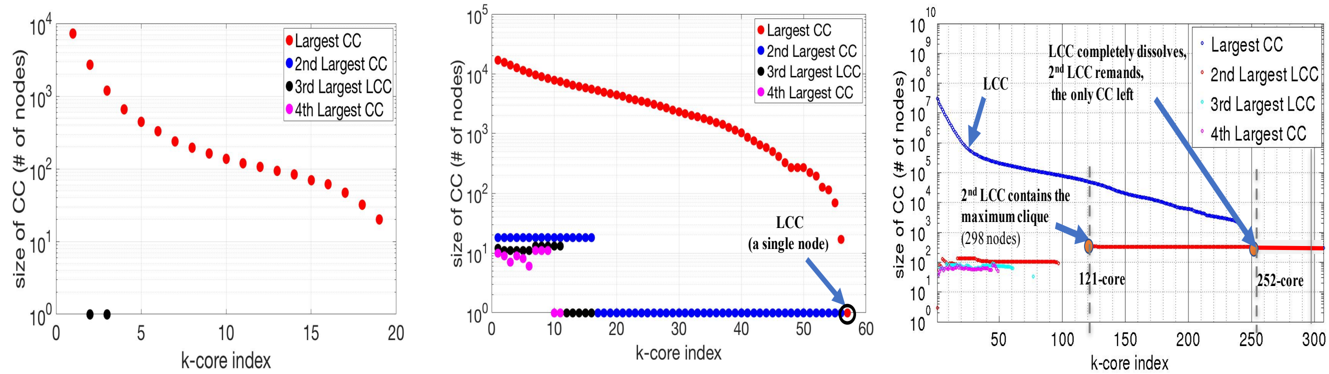


Figure 1: A schematic representation of a network under k-shell decomposition: the network can be viewed as the union of shell 1 up to $k_{max} = 3$ (network core).

Table 1: Main characteristics of the social networks and AS graphs: d - node degree; % LCC - percentage size of the largest connected component of the original network

ID	# nodes	# edges	max(d)	% LCC
arenas-jazz	198	2,742	100	1.00
dnc-corecipient	906	20,858	368	0.94
arenas-pgp	10,680	24,316	205	1.00
Oregon-1	11,174	23,409	2,389	1.00
ca-HepPh	12,008	118,521	491	0.93
ca-AstroPh	18,722	198,110	504	0.95
ca-CondMat	23,133	93,497	280	0.92
email-Enron	36,692	183,831	1,383	0.92
loc-brightkite	58,228	214,078	1,134	0.97
Facebook	63,731	817,035	1,098	0.99

- *ca-AstroPh, ca-HepPh, ca-CondMat*: collaboration networks between authors for papers submitted to Astro Physics, High Energy Physics (Phenomenology category) and Condense Matter Physics – a graph contains an undirected edge (i, j), if author i co-authored a paper with author j.
- *arenas-jazz*: collaboration network between jazz musicians – the graph contains an undirected edge (i, j), if two musicians have played together in a band.
- *email-Enron*: email communication network – the graph contains an undirected edge (i, j), if address i sent at least one email to address j.
- *arenas-pgp*: interaction network of users of the Pretty Good Privacy (PGP) algorithm.
- *dnc-corecipient*: online contact network for people having received the same email in the 2016 Democratic National Committee email leak – the graph contains an undirected edge (i, j), if two persons received the same email.
- *Facebook*: an undirected subgraph of the friendship network for the users in Facebook.
- *loc-brightkite*: an undirected graph for the friendship network for the users from loc-brightkite location-based online social network.



(a) Oregon-1 (b) ca-AstroPh (c) Google+

Figure 2: The size of the largest as well as those of the 2nd, 3rd and 4th largest connected components in the k-core subgraphs

3 K-SHELL NETWORK CORE

K-shell decomposition [15] is one of the most popular and scalable method to investigate and visualize the core-periphery structure in complex networks. This method assigns to each node an integer representing its coreness location according to successive layers or shells in the network. It works as follows: a) first, remove all nodes in the network with degree 1 (and their respective edges) – these nodes are assigned to the 1-shell; b) more generally, at step $k = 2, \ldots$, remove all nodes in the remaining network with degree k or less (and their respective edges) – these nodes are assigned to the k-shell; and c) the process stops when all nodes are removed at the last step. Small values of k define the periphery of the network and the *innermost network core* corresponds to the highest shell index (k_{max}) – see Fig. 1. (Note that this is distinct from *k-core decomposition*[1] defined in the literature [7, 8]).

In the k-shell decomposition process, at each step k, the remaining subgraph is referred to as "k-core" (C_k). The k-core subgraph is the union of all shells with indices larger or equal to k or it is the maximal induced subgraph $C_k \subseteq G$ such that if $v \in C_k$, then node v must have at least $k+1$ neighbors that belong to C_{k-1} and $deg^k(v) > 0$ (we use $deg(v)$ to denote the degree of v in the network and $deg^k(v)$ to denote the degree of v in C_k). Similarly, k-shell (S_k) can be defined as the subgraph induced by the set of nodes with $d^{k-1}(v) \leq k$ and if $v \in S_k \rightarrow deg^k(v) = 0$.

Clearly, for a node to belong to the k-core (thus $shell(v) \geq k$), it must have at least degree k, i.e., $deg(v) \geq k$. However, $deg(v) \geq k$ is not sufficient to guarantee it to belong to the k-core. For example, a node v with only neighbors of degree 1 (i.e., v is the root of a star structure) belongs to the 2-shell, i.e., $shell(v) = 2$, no matter how high its degree is. On the other hand, it is easy to see that if a node v is part of a clique of k nodes, then $shell(v) \geq k$. However, a node v does not need to be part of a k-clique to have $shell(v) \geq k$. Consider a tree T of n nodes (the sparsest graph with n nodes). We can in fact provide a complete characterization of nodes in T to have $shell(v) \geq k$ in a recursive manner: for v to have $shell(v) \geq k$, it must have at least k-neighbors u's with $shell(u) \geq k-1$ – this characterization also applies to a general graph. We see that in the case of a tree, nodes with higher k-shell indices must lie more at the "core" (i.e., the increasingly "denser" part) of the tree. For a general graph, however, a node with a high k-shell index may not lie at the "core" of the graph: it can be part of a large clique that is "isolated" on a periphery of a massive graph. In such a case, the large clique will break off from the "core" of the network (e.g., as represented by the largest connected component remaining in the k-core) in the early stage of the k-shell decomposition process.

This method has been successfully used as a visualization tool for studying and uncovering the core structure of networks such as the Internet AS graph [15]. We apply it to the *Oregon-1* AS dataset. Fig. 2(a) shows the size of the largest as well as those of the 2nd, 3rd and 4th largest connected components in the k-core graph. We observe that the largest connected component decreases smoothly as k varies from 1 to 20. At $k_{max} = 20$, we are left with a very dense core subgraph composed of 20 nodes and 164 edges – the network nucleus. This result shows that for the AS graph, nodes with the highest k-shell indices indeed lie at the "core" (i.e., the increasingly "denser" part) of the graph. However, our experiments reveal that applying the k-shell decomposition for other types of graphs, especially social graphs, may not yield the same results. There are two possible reasons:

First, for some graphs the k_{max}-shell seems to contain some "residual" portions of the nucleus of a graph or simply a singleton node. For example, Fig. 2(b) shows the k-core graph for the 4 largest connected components in the *ca-AstroPh* dataset. We see that at k_{max}=57, we are left with just a single node in the k-core graph, which is unlikely to be the complete inner-core of the graph.

Second, in other graphs the k_{max}-shell does not appear to lie at the "core" of the graph: it could be part of a large community structure (e.g. a maximum clique) that is "isolated" on a periphery of a graph. To illustrate this, we apply the k-shell decomposition method to a *Google+* reciprocal network[2] obtained from a previous

[1]Which simply removes all nodes with degree less than k in a graph.

[2]A network composed with only bi-directional edges, extracted from a directed social graph. A reciprocal network can be viewed as the stable "skeleton" network of a directed social network that holds it together and encodes its main topological characteristics [20]. For more on the reciprocal network of Google+ the reader is referred to [19, 20].

study [19, 20] - it consists of more than 40 million nodes and ≈ 400 million edges. Figure 2(c) shows the size of the largest as well as those of the 2nd, 3rd and 4th largest connected components in the k-core, as k varies from 1 to 308. We note that at step $k = 121$, a small subgraph containing the maximum clique (of size 290) breaks off from the largest connected component which desolves after $k = 253$, whereas this subgraph containing the maximum clique persists after $k = 252$ and becomes the largest component; at $k_{max} = 308$, we are left with this maximum clique plus 10 additional nodes that are connected to the maximum clique. Closer inspection of the nodes in the maximum clique reveals that its users belong to a single institution in Taiwan, forming a close-knit community where each user follows everyone else – which is unlikely to be the network core of Google+.

From these results, we see that directly applying the standard k-shell decomposition to some graphs (especially, social networks) produces an "innermost" structure that does not represent "core" of these networks. This is due to the fact that at a certain k-index, we reach the vicinity of the core; but going far beyond this index would destroy the core structure of the network.

4 NODE DEPENCENCE VALUES AND NETWORK CORE

In order to extract a meaningful "core" for a general graph $G = (V, E)$ (e.g., social networks), we therefore modify the standard k-shell decomposition method to stop the process earlier. To achieve this, we propose a new metric that provides important information about the structural function of each node in the graph (we label it as "dependence" value) at each k-step. Then, we present a new measure called *nucleon-index* (NI) that captures the extend to which a subgraph is a densely intra-connected and topological central core – it can be used with a wide variety of functions to transition between core and peripheral nodes (e.g., dependence value, closeness and betweenness centralities, etc).

4.1 Node Depencence Values

The *dependence* value of node v at step k is defined as follows: for $v \in V$, $dep^0(v, \beta) = 0$ and for $k = 1, \ldots, c(v)$,

$$dep^k(v, \beta) := dep^{k-1}(v, \beta) + \delta^k(v) + \beta \times \Sigma_{u \in N^k(v)}[dep^{k-1}(u, \beta)] \tag{1}$$

where β is a control parameter, $0 \leq \beta \leq 1$; $N^k(v)$ is the set of neighbors of node v that are removed at step k, and $\delta^k(v) = |N^k(v)|$. The dependency of node v is recursively defined by measuring the number of nodes u (the h-hop neighbors of v, $h = 1, \ldots, k$) that are removed in earlier steps up to $k = c(v)$ –the *coreness* of node v (and for $k \geq c(v)$, by convention, we define $dep^k(v, \beta) = dep^{c(v)}(v, \beta)$).

Intuitively, $dep^k(v, \beta)$ captures the number of nodes recursively dependent on v that have been removed in earlier steps up to k. With $\beta = 0$, we note that $dep^k(v, \beta)$ captures the number of v's neighbors removed at each step up to k, and for $k \geq c(v)$, $dep^k(v, \beta) = \sum_k \delta^k(v) = deg(v)$, the degree of node v. With $\beta > 0$, $dep^k(v, \beta)$ captures not simply the dependence of its neighbors, but that of its neighbors' neighbors, and so forth. However, the number of nodes u removed at each step up to k does not influence the

Table 2: *Arenas – jazz*: **peak nucleon-indices (*NI*) and their respective k_C-indices (set *SK*) and β values**

β	max(NI)	k_C
0.0	0.011019	26
0.1	0.006561	25
0.2	0.006125	24
0.3	0.006841	24
0.4	0.007256	24
0.5	0.007500	24
0.6	0.007818	25
0.7	0.008545	25
0.8	0.009222	25
0.9	0.009849	25
1.0	0.010433	25

dependence value of the node v uniformly. Their contribution is weighted by the parameter β in eq.(1). The parameter β quantifies the contribution of node u to the total dependence value of node v. More precisely, at the kth-step, we multiply the number of h-step removed neighbors of v by β^{h-1} (see the proof in the appendix). Thus, the further a node u is to node v, the less it will contribute to the total dependence value of node v. Hence, a node v having more nodes u with high dependence values in its vicinity will also have a high dependence value, creating the *dependency propagation* effect. Therefore, we posit that the network core should contain only nodes with very high dependence because the $dep^k(v, \beta)$ values of any $v \in V$ grows as k increases (more nodes are removed as we move from the periphery of the graph to its core). In the next section, we use the dependence value of node v as a measure of its coreness.

4.2 Nucleon Index and Network Nucleus

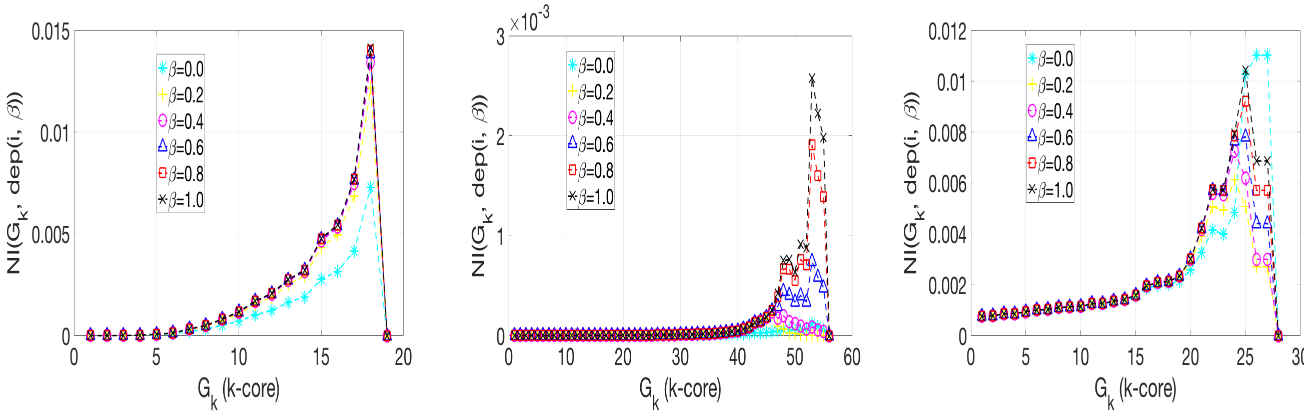
To derive a meaningful "core" structure in social networks, we postulate that the *nucleus* of a network $G(V, E)$ is an induced subgraph G_C having the following properties:

(1) Subgraph $G_C(V_C, E_C)$ is *connected* and composed of a collection of nodes in G with *dense* aggregate centralities by some measure.
(2) The set V_C is fundamental for the *structural properties* of the network, e.g., in terms of connecting nodes via short paths through the network.
(3) G_C is the minimal subgraph with these properties.

To find a subgraph G_C with the above properties, we consider an appropriately defined "decomposition" process (e.g., the k-shell decomposition) which yields a (filtration) sequence of (sub)graphs $\{G_k\}$'s of G: $G_0 := G \supset G_1 \supset \cdots \supset G_K = \emptyset$. Given a node centrality measure $\theta(i)$, $i \in V$, we define the *nucleon-index* (NI) to capture the extent to which a subgraph constitutes a "densely connected", topological central core in this sequence:

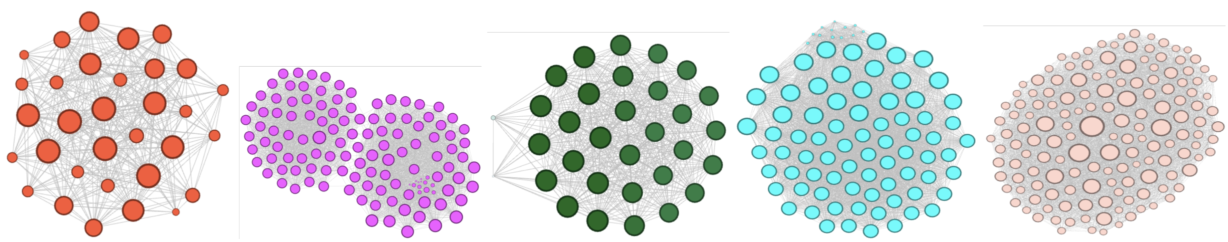
$$NI(G_k, \theta(i)) := \frac{V_k}{V_{k-1}} \times \frac{E_k}{V_k \times (V_k - 1)} \times \{\frac{1}{V_k} \times \sum_{i \in G_k} \theta(i)\} \tag{2}$$

where by abuse of notation, we use E_k to denote the number of edges between nodes in G_k and V_k the number of nodes in G_k (and $|V_K| = 0$). The second term in eq.(2) measure the density of G_k and the last term the average centrality of G_k. Ideally, if G_k



(a) Oregon-1 (b) ca-AstroPh (c) arenas-jazz

Figure 3: Variation of the nucleon-index per k-core index for several β parameters in the dependence computation



(a) Oregon-1 (b) ca-AstroPh (c) arenas-jazz (d) dnc-corecipient (e) email-Enron

Figure 4: Visualization of the core subgraphs: the size of a node is proportional to its degree

is a "dense core" of G, the product of these two terms should be large. The first term controls the rate of changes in size from G_k to G_{k+1}: intuitively, if G_k is the "nucleus" of G, going from G_{k-1} to G_k should not drastically change its size; but going from G_k to G_{k+1} amounts to breaking G_k apart, yielding a collection of small connected components. In other words, V_{k+1} would fall off quickly, as G_{k+1} is a small connected subgraph or an empty graph. Hence, G_k with the largest *NI* represents the *nucleus* of G (as produced by the decomposition process).

Considering the node dependence value as a centrality measure, we define $\theta(i)$ as follows:

$$\theta(i) := \frac{dep^{c(i)}(i, \beta)}{\sum_{j \in G} dep^{c(j)}(j, \beta)}. \quad (3)$$

Using $\theta(i)$ defined above and applying the nucleon-index to the k-shell decomposition procedure, we develop the following *stop rule* for core extraction.

Stopping rule for core extraction: For any graph G with a dense core structure, we should stop the k-shell decomposition method at the induced subgraph of the k_C-core with maximal nucleon-index. Thus, we seek a k_C-index that maximizes the nucleon-index (NI).

Figure 3 plots the nucleon-indices per k-core (C_k) for Oregon-1, ca-AstroPh and arenas-jazz networks. To select the optimal β parameter for eq. (1), we use the following criteria: let's assume that SK is the set of the k-indices corresponding to the maximum nucleon-indices, as β varies in the interval [0, 1] and k increases from 1 up to k_{max}. Then, we select any β associated with the k-index which appears most often in the set SK. For example, Table 2 shows the set SK for arenas-jazz. We select a β corresponding to the mode k_C-index value of 25 (i.e., $\beta = 0.1$; $\beta = 0.6$; $\beta = 1.0$).

Table 3 shows the (k_{max}, β, k_C) indices for our social network and Internet AS datasets and Fig. 4 provides a visualization of our extracted core subgraphs (G_C) for several example networks[3]. The smallest subgraph has 32 nodes and 362 edges (Oregon-1), whereas the largest one has 239 nodes and 28,441 edges (ca-HepPh). We will further investigate the structure of these core subgraphs (network nuclei) in the remaining sections.

4.3 Other Centralities and Nucleus

Nodes are more likely to be part of a network's core if they have high centrality score and if they are connected to other core nodes. Equation (2) can be used with a wide variety of $\theta(i)$ functions to transition between core and peripheral nodes. Thus, it allows one to use different ways to compute the nucleon-index (NI) and measure core quality. Here, we compute the nucleon-index using some of the most common centrality metrics: closeness centrality (c_c) [41,

[3]We omit the others plots here due to space constraint.

Table 3: maximum k-shell index (k_{max}); β parameter; k-index to stop the shells pruning process (k_C); number of nodes and edges in the core subgraph N(G_C) and E(G_C)

Network	k_{max}	β	k_C	$N(G_C)$	$E(G_C)$
arenas-jazz	29	0.6	25	32	466
dnc-corecipient	75	0.5	67	87	3,118
arenas-pgp	33	0.5	31	38	658
Oregon-1	20	0.5	18	32	362
ca-HepPh	238	0.5	99	239	28,441
ca-AstroPh	57	0.6	53	126	3,378
ca-CondMat	51	0.5	37	37	382
email-Enron	51	0.5	48	150	4,395
loc-brightkite	58	0.5	56	66	1,893
Facebook	64	0.5	61	285	9,616

Table 4: k-index to stop the shells pruning process (k_C) for several centralities: c_c - closeness centrality; b_c - betweenness centrality; e_c - eigenvector centrality; dep - dependence

Network	k_C			
	$\theta(i) = c_c$	$\theta(i) = b_c$	$\theta(i) = e_c$	$\theta(i) = dep$
arenas-jazz	26	25	26	25
dnc-corecipient	68	65	68	67
arenas-pgp	31	30	31	31
Oregon-1	18	18	18	18
ca-HepPh	99	99	99	99
ca-AstroPh	53	53	53	53
ca-CondMat	42	37	37	37
email-Enron	48	48	48	48
loc-brightkite	55	48	56	56
Facebook	60	60	60	61

42, 45], betweenness centrality (b_c) [14, 41, 45] and eigenvalue centrality (e_c) [12, 34, 41, 45] – we compare the obtained k_C-indices with the values computed in the previous section.

The closeness centrality measures how central a node is in terms of its distance (shortest path) from all other nodes [41], while the betweenness centrality for a node measures the number of shortest paths that pass through that node [41]. The eigenvalue centrality computes the centrality for a node based on the centrality of its neighbors. It is based on the notion that a node should be viewed as important if it is linked to other important nodes, where a node importance (or centrality score) corresponds to the largest eigenvector of the adjacency matrix [41]. Table 4 shows the k_C-indices for the different centrality measures and Fig. 5 plots the nucleon-indices versus k-core indices of several example networks[4]. In general, we observe that all the centralities give consistent k_C-indices or core structures for our datasets. In particular, we observe that our dependence metric, $dep(i, \beta)$, derives similar core structure when compared to the other metrics. From the consistency of the results given by the studied centrality metrics, we can infer that our social networks (see § 2) truly have a core structure.

[4]We omit the others plots here due to space constraint.

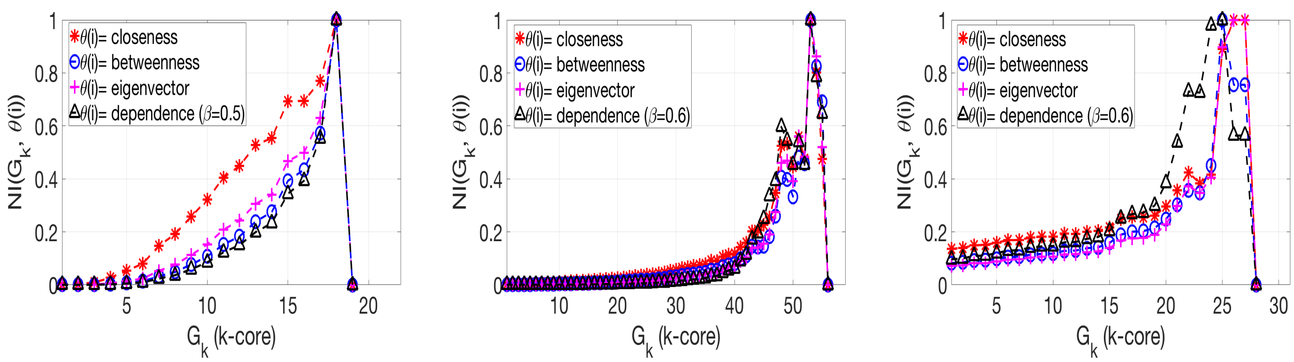
Table 5: Comparing classical k-shell decomposition (KS), Nucleon Index (NI) + k-shell decomposition (KS) and Rich-Club network core (G_C) in real-world networks : N - number of nodes; E - number of edges; D - diameter; P - path length; ρ - density

method	dataset	N	E	D	P	ρ
Classical KS	Oregon-1	20	164	2.0	1.14	0.86
	ca-AstroPh	17	136	1.0	1.00	1.00
	email-Enron	36	472	2.0	1.25	0.75
NI + KS	Oregon-1	32	363	2.0	1.27	0.73
	ca-AstroPh	126	3,378	3.0	1.87	0.43
	email-Enron	150	4,395	3.0	1.61	0.39
Rich-Club	Oregon-1	37	314	3.0	1.57	0.47
	ca-AstroPh	82	994	3.0	1.80	0.30
	email-Enron	106	1,660	4.0	1.77	0.30

All the centrality metrics discussed here are designed to measure notions of node importance in a network. Nevertheless, they have different computational complexity and require different network information. For example, the closeness and eigenvalue centralities need the full network information and have a high complexity of O(V^3). The betweenness centrality has a lower complexity of $O(VE)$ [14]. Our approach to calculate the $dep(v, \beta)$ score for node v is dependent on the k-shell decomposition method and degree computation which have a complexity of $O(V+E)$. Then, given that the degree and coreness of each node are known, our procedure has a complexity of $O(E)$. For a large sparse social network with $O(n)$ edges, this yields a linear time algorithm. Therefore, our methodology is highly scalable and can be applied to massive networks (hundreds million nodes and billion edges).

We compare our methodology to extract core subgraphs to the classical k-shell decomposition [15] and rich club [31, 51] methods. Table 5 provides statistics for the structure of the derived core subgraphs (G_C) for three of our networks (i.e., *Oregon-1, ca-AstroPh* and *email-Enron*) – we omit the others networks here due to space constraint. In general, for our dataset, we observe that the classical k-shell decomposition method (KS) is bias toward small and highly dense core subgraphs, G_C^{KS}, (i.e., a clique) which may not represent the “network core" (see § 3). In contrast, our modified k-shell decomposition method ($NI + KS$) generates larger core subgraphs than KS. In fact, our core subgraphs are supersets of the cores extracted using KS: $G_C^{NI+KS} \supset G_C^{KS}$. When compared to rich-club, we see that for some networks our modified k-shell decomposition method ($NI + KS$) generates core subgraphs of similar size (e.g., *Oregon-1*). However, our core subgraphs have more compact structure: small diameter, small path length and high density. For other networks, our methodology generates larger and denser core subgraphs than the rich-club method (e.g., *email-Enron*). This can be explained due to the fact that the rich-club is bias toward nodes with higher degree[5]. Differently, our definition of core is more general, and it allows low-degree nodes to belong to the core, as long as, they are important components in the structure of the network.

[5]Rich-club is a group of high-degree nodes in a network that preferentially connect to one another. This structure might be the core subgraph for power law networks



(a) Oregon-1 (b) ca-AstroPh (c) arenas-jazz

Figure 5: Variation of the nucleon-index (NI) per k-core index for several centrality metrics: the value of NI is normalized; the k-index to stop the shells pruning process (k_C) corresponds to the max(NI)

Table 6: Summary of path length (P) and diameter (D) characteristics: $\delta(u, G_C)$ - shortest path from node u to the core subgraph G_C

Network	P	D	$Avg(\delta(u, G_C))$
arenas-jazz	2.21	6	1.27
dnc-corecipient	2.27	8	1.63
arenas-pgp	7.65	24	4.27
Oregon-1	3.62	10	1.54
ca-HepPh	4.67	13	2.38
ca-AstroPh	4.17	14	2.24
ca-CondMat	5.35	14	3.25
email-Enron	4.03	13	1.74
loc-brightkite	4.92	18	3.41
Facebook	4.31	15	2.42

5 ANALYSIS OF THE NETWORK CORE STRUCTURE

Given the dense structures of our core subgraphs, illustrated in Figure 4, we now investigate the importance of this substructure for the network. To achieve this, we define and analyse the following metrics:

Core Path Length: To understand how much the network core contributes towards the small path lengths, we measure how many hops there are between any user to the core subgraph: $\delta(u, G_C) = min_{y \in G_C}\{d(u, y)\}$; $G_C \subset G$. Figure 6 presents the core path length and network path length distribuitions for *Oregon-1*, *ca-AstroPh* and *arenas-jazz*[6], whereas Table 6 shows the average values and the diameter for all the networks. From these results, we can see that most users are approximately 4 hops away from a random user and at most 2 hops away from the core (G_C), which implies that our core subgraphs are important structure for the connectivity of the nodes in the network.

[6]We obtain similar results for the other datasets. We omit the plots here due to space constraint.

Table 7: Ratio of the distance between nodes u and v to their respective distance to the core subgraph G_C: $R(u, v)$

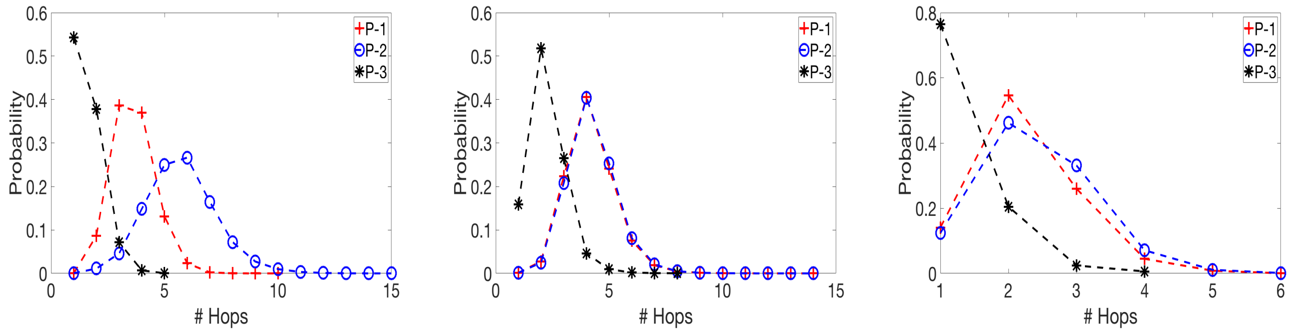
Network	k	Avg($R(u, v)$)
arenas-jazz	70	0.96
dnc-corecipient	700	0.90
arenas-pgp	8,000	0.89
Oregon-1	8,000	1.21
ca-HepPh	8,000	1.03
ca-AstroPh	8,000	0.96
ca-CondMat	20,000	0.84
email-Enron	20,000	1.21
loc-brightkite	20,000	0.73
Facebook	20,000	0.92

Core Centrality: We now investigate the importance of the core subgraph for communication and information diffusion in the network. To achieve this, we use the following procedure: first, we randomly sample k unique pairs of nodes (u, v). Then, we measure, $R(u, v)$, the ratio of the distance between nodes u and v to their respective distance to the core subgraph, as expressed in eq.(4), where $d(u, v)$ represents the shortest path between u and v, and $d(u, G_C)$ or $d(v, G_C)$ represents the shortest path between u or v to the core subgraph G_C.

Table 7 shows the average $R(u, v)$ for $k = 70$, $k = 700$, $k = 8,000$ and $k = 20,000$ respectively. We observe that the avg($R(u, v)$) is very close to the optimal value of 1.0, which implies that our core subgraph G_C contains the nodes with the highest *betweeness* in the network and they act as "bridges" for the connectivity between the other nodes in the network.

$$R(u, v) = \frac{d(u, v)}{d(u, G_C) + d(v, G_C)} \quad (4)$$

Core Removal: Lastly, we investigate the impact of removing the core subgraph G_C in the structure of the studied networks. We observe that all the neworks described in § 2 have a giant connected component (GCC) contaning more than 90% of all the nodes and more than 85% of all edges in the network. After the core removal,



(a) Oregon-1 (b) ca-AstroPh (c) arenas-jazz

Figure 6: Path length distribuitions: P-1: distance between nodes in the original network; P-2: distance between nodes in the original network, after core removal; P-3: nodes distance to the core subgraph G_C

we see that, for some networks (i.e., arenas-jazz, dnc-corecipient, Oregon-1 and email-Enron), at least 20% of the nodes break away from GCC, forming many isolated components of smaller sizes. Table 8 shows the number of these new connected components per network as well as the ratio of the size of the GCC after and before call removal in terms of the number of nodes and edges. From these results, we deduce that removing G_C significantly affects the connectivity and density for some of the networks.

Figure 6 shows the path length distribuition after we remove the core from our networks. We observe that the average path length increases after the core removal for most of the networks. For example, *ca-AstroPh*, *email-Enron* and *Oregon-1* have average path length of 4.17, 4.03 and 3.62 before core removal, and 4.25, 4.49 and 5.72 after core removal. This result provides further evidence that the core subgraph G_C is an important structure for reachability, communication and information diffusion in these networks. Next, we discuss the implications of our results.

Table 8: Basic stats of the giant (largest) connected components (GCC) after core removal: c_n - number of connected components; n_j and n_i- number of nodes in GCC before and after core removal; e_j and e_i - number of edges in GCC before and after core removal

Network	# c_n	n_i/n_j	e_i/e_j
arenas-jazz	2	0.833	0.612
dnc-corecipient	104	0.757	0.404
arenas-pgp	26	0.993	0.940
Oregon-1	3,183	0.688	0.503
ca-HepPh	73	0.967	0.645
ca-AstroPh	12	0.946	0.929
ca-CondMat	2	0.997	0.978
email-Enron	3,350	0.800	0.711
loc-brightkite	65	0.972	0.957
Facebook	66	0.994	0.930

6 DISCUSSION

Using examples from communication networks as well as collaboration, location-based, interaction, and online social networks, we have demonstrated that our method can effectively uncover and extract the nucleus of these networks. In this section, we discuss the limitations and implications of our method and results.

First, our proposed methodology to uncover the nucleus of networks can also be applied to weighted and directed networks by using a variation of the k-shell decomposition method: Garas et al. [24] presented a weighted k-shell decomposition method and Batagelj et al. [10] generalized the k-shell decomposition to directed networks. Our method can be applied with these generalized algorithms because our dependence and nucleon-index metrics are independent to the k-shell decomposition method. Once the k-shells are provided by decomposing the network into k-layers, the dependence and nucleon-index values can be computed.

Second, the "coreness" centrality or k-shell index has been argued to be a better measure than node degree for identifying influential spreaders in a network [23, 28]. However, our results show that using k-shell indices as a predictor of spreading influence of a node can be misleading. This is due to the fact that for a node to have a high k-shell index, it just needs to be a part of a very strong structure (e.g., a clique). This structure, however, may be isolated and lie at the edge or periphery of the network, instead of its core (see § 3). Our analysis shows that the dependency value of a node, $dep^k(i)$, provides important information about the structure function of each node in the graph. Thus, we believe that by using a node dependency value along with its k-shell index (dep^k, k), we can better predict the spreading influence of a node than simply using its k-shell index. We will investigate this in the future.

Third, unveiling the core structure of social networks may have implications in the design of algorithms for information flow, and in development of techniques for analysing the vulnerability or robustness of networks. In addition, analysis of the core structure of social networks can help us uncover and understand possible organizing principles shaping the observed network topological structure and network formation.

7 RELATED WORK

In contrast to the wealth of attention given to community structure analysis in the literature, there are comparatively few methods for extracting and analyzing the core structure of a network. Some studies simply define the network "core" as the maximal clique composed of the highest degree nodes in a network [44], while other studies focus instead on some notion of connectivity information (e.g. betweeness, closeness, etc.) to find the core and periphery of a network [16, 17, 27, 33, 43].

One of the most popular quantitative methods to investigate core-periphery structure was proposed by Borgatti and Everett in 1999 [13]. Based on this study, several methods for identifying the core-periphery of a network have been proposed [16, 17, 27]. These algorithms attempt to determine which nodes are part of a densely-connected core and which are part of a sparsely connected periphery by solving some complex optimization problem. Consequently, most of these methods are computationally expensive and do not scalable to large networks.

The authors in [47] used the notion of α-β community to extract the "core" of a graph. An α-β community is a connected subgraph C with each vertex in C connected to at least β vertices of C and each vertex outside of C connected to at most α vertices of C ($\alpha < \beta$). They extract the network core structure by taking the intersection of α-β communities of different size k. A core thus corresponds to one or multiple dense regions of the graph. As a result, the proposed heuristics in [47] may return multiple dense regions ("cores") for a given network. In addition, this algorithm does not guarantee to terminate within a reasonable amount of running time.

8 CONCLUSION

In this paper, we have advanced and developed an effective procedure to extract the *core* structure of social networks. First, we introduce a new metric – the node "dependence value" – that measures the location importance of a node in a network. Second, we define a new measure called *nucleon-index* that captures the extend to which a subgraph is a densely intra-connected and topological central core. Then, using these metrics, we proposed a modified version of the k-shell decomposition method by identifying the k_C-index where we should stop pruning the network in order to preserve its core structure. For our social network datasets, we found that they contain very dense core subgraphs G_C. The smallest core has 32 nodes and 362 edges (Oregon-1), whereas the largest one has 239 nodes and 28,441 edges (ca-HepPh). Finally, given a dense core subgraph G_C, we investigate the importance of this substructure for the network by analysing the following metrics: i) the distance between a node v to the core subgraph G_C; ii) the ratio of the distance between nodes u and v to their respective distance to G_C and iii) lastly, the impact of removing G_C in the structure of the network G ($G_C \subset G$).

As part of ongoing and future work, we will provide a more in-depth analysis of the dense core subgraph G_C of social networks. We also plan to apply our method to a massive Google+ dataset [19, 20, 26] (with more than 170 million nodes and ≈ 3 billion edges), a massive Twitter dataset [21] (with more than 500 million nodes and ≈ 23 billion edges) and other social networks.

ACKNOWLEDGMENTS

This research was supported in part by DoD ARO MURI Award W911NF-12-1-0385, DTRA grant HDTRA1- 14-1-0040, NSF grant CNS-1411636, CNS-1618339 and CNS-1617729.

9 APPENDIX

Beta Parameter Selection: We now establish that the contribution of the h-step removed neighbors of node i is attenuated by β^{h-1}:

Given that $dep^0(i) = 0$ and $dep^1(i) = \delta^1(i)$, we can write an expression for $dep^2(i)$ as following:

$$\begin{aligned} dep^2(i) &= dep^1(i) + \delta^2(i) + \beta \times \Sigma_{j \in N^2(i)} dep^1(j) \\ &= \delta^1(i) + \delta^2(i) + \beta \times \Sigma_{j \in N^2(i)} \delta^1(j) \end{aligned} \tag{5}$$

Let us assume that node i has $c(i) = 4$, then $dep^4(i)$ is computed as following:

$$dep^4(i) = dep^3(i) + \delta^4(i) + \beta \Sigma_{j \in N^4(i)} [dep^3(j)] \tag{6}$$

Expanding eq. (6) yields:

$$\begin{aligned} dep^4(i) = dep^3(i) + \delta^4(i) + \beta \Sigma_{j \in N^4(i)} [dep^2(j) + \delta^3(j) \\ + \beta \Sigma_{j' \in N^3(j)} dep^2(j')] \end{aligned}$$

Substituting eq. (5) yields:

$$\begin{aligned} dep^4(i) := dep^3(i) + \delta^4(i) + \beta \Sigma_j [M^3(j) + \beta \delta^2(j) \rho^1(j'^*) \\ + \beta \Sigma_{j'} [M^2(j') + \beta \delta^2(j') \rho^1(j'')]] \end{aligned}$$

where $M^k(i) = \Sigma_k \delta^k(i)$ and $\delta^k(i) = \rho^k(i), \forall i \in V$.

Further simplify $dep^4(i)$ yields:

$$\begin{aligned} dep^4(i) := dep^3(i) + \delta^4(i) + \Sigma_j [\beta M^3(j) + \beta^2 \delta^2(j) \rho^1(j'^*) \\ + \Sigma_{j'} [\beta^2 M^2(j') + \beta^3 \delta^2(j') \rho^1(j'')]] \end{aligned}$$

We can rewrite the above expressions as:

$$dep^4(i) := dep^3(i) + \beta^0 A + \Sigma_j [\beta B + \beta^2 C + \Sigma_{j'} [\beta^2 D + \beta^3 E]] \tag{7}$$

where:

- $A = \delta^4(i)$: 1-step neighbors of i removed at $k = 4$
- $B = M^3(j)$: 2-step neighbors of i removed at $k = 1, 2, 3$
- $C = \delta^2(j) \rho^1(j'^*)$: 3-step neighbors of i removed at $k = 1$
- $D = M^2(j')$: 3-step neighbors of i removed at $k = 1, 2$
- $E = \delta^2(j') \rho^1(j'')$: 4-step neighbors of i removed at $k = 1$

By generalizing eq. (7) ($k = 5, ..., n$), we observe that at every k-index, the number of h-step removed neighbors of i is multiplied by β^{h-1}. This concludes our proof. As stated before, the parameter β quantifies the contribution of node j to the total dependence value of node i. Thus, by varying β, we are impacting the contribution of any node j to the total dependence value of node i by the same proportion.

REFERENCES

[1] [n. d.]. Pretty Good Privacy network dataset – KONECT.
[2] [n. d.]. Stanford Large Network Dataset Collection. https://snap.stanford.edu/data/.
[3] 2016. DNC emails co-recipients network dataset – KONECT. http://konect.uni-koblenz.de/networks/dnc-corecipient.
[4] 2016. Facebook friendships network dataset – KONECT. http://konect.uni-koblenz.de/networks/facebook-wosn-links
[5] 2016. Jazz musicians network dataset – KONECT. http://konect.uni-koblenz.de/networks/arenas-jazz.
[6] Yong-Yeol Ahn, James P Bagrow, and Sune Lehmann. 2010. Link communities reveal multiscale complexity in networks. *Nature* 466, 7307 (2010), 761–764.
[7] José Ignacio Alvarez-Hamelin, Luca Dall'Asta, Alain Barrat, and Alessandro Vespignani. 2005. K-core decomposition of internet graphs: hierarchies, self-similarity and measurement biases. *arXiv preprint cs/0511007* (2005).
[8] J Ignacio Alvarez-Hamelin, Luca Dall'Asta, Alain Barrat, and Alessandro Vespignani. 2006. Large scale networks fingerprinting and visualization using the k-core decomposition. In *Advances in neural information processing systems*. 41–50.
[9] Reid Andersen, Fan Chung, and Kevin Lang. 2006. Local graph partitioning using pagerank vectors. In *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*. IEEE, 475–486.
[10] Vladimir Batagelj and Matjaž Zaveršnik. 2002. Generalized cores. *arXiv preprint cs/0202039* (2002).
[11] MariÃąn BoguÃśÃą, Romualdo Pastor-Satorras, Albert DÃŋaz-Guilera, and Alex Arenas. 2004. Models of Social Networks based on Social Distance Attachment. *Phys. Rev. E* 70, 5 (2004), 056122.
[12] Phillip Bonacich. 1987. Power and centrality: A family of measures. *American journal of sociology* 92, 5 (1987), 1170–1182.
[13] Stephen P Borgatti and Martin G Everett. 2000. Models of core/periphery structures. *Social networks* 21, 4 (2000), 375–395.
[14] Ulrik Brandes. 2001. A faster algorithm for betweenness centrality. *Journal of mathematical sociology* 25, 2 (2001), 163–177.
[15] Shai Carmi, Shlomo Havlin, Scott Kirkpatrick, Yuval Shavitt, and Eran Shir. 2007. A model of Internet topology using k-shell decomposition. *Proceedings of the National Academy of Sciences* 104, 27 (2007), 11150–11154.
[16] Marcio Rosa Da Silva, Hongwu Ma, and An-Ping Zeng. 2008. Centrality, network capacity, and modularity as parameters to analyze the core-periphery structure in metabolic networks. *Proc. IEEE* 96, 8 (2008), 1411–1420.
[17] Fabio Della Rossa, Fabio Dercole, and Carlo Piccardi. 2013. Profiling core-periphery network structure by random walkers. *Scientific reports* 3 (2013), 1467.
[18] Patrick Doreian. 1985. Structural equivalence in a psychology journal network. *Journal of the American Society for Information Science* 36, 6 (1985), 411–417.
[19] Braulio Dumba, Golshan Golnari, and Zhi-Li Zhang. 2016. Analysis of a Reciprocal Network Using Google+: Structural Properties and Evolution. In *International Conference on Computational Social Networks*. Springer, 14–26.
[20] Braulio Dumba and Zhi-Li Zhang. 2016. Unfolding the Core Structure of the Reciprocal Graph of a Massive Online Social Network. In *International Conference on Combinatorial Optimization and Applications*. Springer, 763–771.
[21] Maksym Gabielkov, Ashwin Rao, and Arnaud Legout. 2014. Studying social networks at scale: macroscopic anatomy of the twitter social graph. In *ACM SIGMETRICS Performance Evaluation Review*, Vol. 42. ACM, 277–288.
[22] Jing Gao, Feng Liang, Wei Fan, Chi Wang, Yizhou Sun, and Jiawei Han. 2010. On community outliers and their efficient detection in information networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 813–822.
[23] Antonios Garas, Panos Argyrakis, Céline Rozenblat, Marco Tomassini, and Shlomo Havlin. 2010. Worldwide spreading of economic crisis. *New journal of Physics* 12, 11 (2010), 113043.
[24] Antonios Garas, Frank Schweitzer, and Shlomo Havlin. 2012. A k-shell decomposition method for weighted networks. *New Journal of Physics* 14, 8 (2012), 083030.
[25] Pablo M. Gleiser and Leon Danon. 2003. Community Structure in Jazz. *Advances in Complex Systems* 6, 4 (2003), 565–573.
[26] Roberto Gonzalez, Ruben Cuevas, Reza Motamedi, Reza Rejaie, and Angel Cuevas. 2013. Google+ or google-?: dissecting the evolution of the new osn in its first year. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 483–494.
[27] Petter Holme. 2005. Core-periphery organization of complex networks. *Physical Review E* 72, 4 (2005), 046111.
[28] Maksim Kitsak, Lazaros K Gallos, Shlomo Havlin, Fredrik Liljeros, Lev Muchnik, H Eugene Stanley, and Hernán A Makse. 2010. Identification of influential spreaders in complex networks. *Nature physics* 6, 11 (2010), 888–893.
[29] JÃĪrÃťme Kunegis. 2013. KONECT – The Koblenz Network Collection. In *Proc. Int. Conf. on World Wide Web Companion*. 1343–1350. http://userpages.uni-koblenz.de/~kunegis/paper/kunegis-koblenz-network-collection.pdf
[30] Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. 2008. Statistical properties of community structure in large social and information networks. In *Proceedings of the 17th international conference on World Wide Web*. ACM, 695–704.
[31] Julian J McAuley, Luciano da Fontoura Costa, and Tibério S Caetano. 2007. Rich-club phenomenon across complex network hierarchies. *Applied Physics Letters* 91, 8 (2007), 084103.
[32] Daniele Miorandi and Francesco De Pellegrini. 2010. K-shell decomposition for dynamic complex networks. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2010 Proceedings of the 8th International Symposium on*. IEEE, 488–496.
[33] Alan Mislove, Massimiliano Marcon, Krishna P Gummadi, Peter Druschel, and Bobby Bhattacharjee. 2007. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 29–42.
[34] Mark Newman. 2010. *Networks: an introduction*. Oxford university press.
[35] Mark EJ Newman. 2004. Detecting community structure in networks. *The European Physical Journal B-Condensed Matter and Complex Systems* 38, 2 (2004), 321–330.
[36] Mark EJ Newman. 2004. Fast algorithm for detecting community structure in networks. *Physical review E* 69, 6 (2004), 066133.
[37] Mark EJ Newman. 2006. Finding community structure in networks using the eigenvectors of matrices. *Physical review E* 74, 3 (2006), 036104.
[38] Spiros Papadimitriou, Jimeng Sun, Christos Faloutsos, and S Yu Philip. 2008. Hierarchical, parameter-free community discovery. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 170–187.
[39] M Puck Rombach, Mason A Porter, James H Fowler, and Peter J Mucha. 2014. Core-periphery structure in networks. *SIAM Journal on Applied mathematics* 74, 1 (2014), 167–190.
[40] Martin Rosvall and Carl T Bergstrom. 2007. An information-theoretic framework for resolving community structure in complex networks. *Proceedings of the National Academy of Sciences* 104, 18 (2007), 7327–7331.
[41] Diego F Rueda, Eusebi Calle, and Jose L Marzo. 2017. Robustness comparison of 15 real telecommunication networks: Structural and centrality measurements. *Journal of Network and Systems Management* 25, 2 (2017), 269–289.
[42] Gert Sabidussi. 1966. The centrality index of a graph. *Psychometrika* 31, 4 (1966), 581–603.
[43] Murray Shanahan and Mark Wildie. 2012. Knotty-centrality: finding the connective core of a complex network. *PLoS One* 7, 5 (2012), e36579.
[44] Georgos Siganos, Sudhir Leslie Tauro, and Michalis Faloutsos. 2006. Jellyfish: A conceptual model for the as internet topology. *Journal of Communications and Networks* 8, 3 (2006), 339–350.
[45] Lei Tang and Huan Liu. 2010. Community detection and mining in social media. *Synthesis lectures on data mining and knowledge discovery* 2, 1 (2010), 1–137.
[46] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P. Gummadi. 2009. On the Evolution of User Interaction in Facebook. In *Proc. Workshop on Online Social Networks*. 37–42.
[47] Liaoruo Wang, John Hopcroft, Jing He, Hongyu Liang, and Supasorn Suwajanakorn. 2013. Extracting the core structure of social networks using (α, β)-communities. *Internet Mathematics* 9, 1 (2013), 58–81.
[48] Bo Wei, Jie Liu, Daijun Wei, Cai Gao, and Yong Deng. 2015. Weighted k-shell decomposition for complex networks based on potential edge weights. *Physica A: Statistical Mechanics and its Applications* 420 (2015), 277–283.
[49] Jaewon Yang and Jure Leskovec. 2012. Structure and overlaps of communities in networks. *arXiv preprint arXiv:1205.6228* (2012).
[50] Tianbao Yang, Rong Jin, Yun Chi, and Shenghuo Zhu. 2009. Combining link and content for community detection: a discriminative approach. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 927–936.
[51] Shi Zhou and Raúl J Mondragón. 2004. The rich-club phenomenon in the Internet topology. *IEEE Communications Letters* 8, 3 (2004), 180–182.