Proceedings of the Ninth International AAAI Conference on Web and Social Media

DUKE: A Solution for Discovering Neighborhood Patterns in Ego Networks

Syed Agha Muhammad
Embedded Sensing Systems
Technische Universität Darmstadt, Germany
muhammad@rbg.informatik.tu-darmstadt.de

Kristof Van Laerhoven
Embedded Systems
Universität Freiburg, Germany
kristof@ese.uni-freiburg.de

Abstract

Given the rapid growth of social media websites and the ease of aggregating ever-richer social data, an inevitable research question that can be expected to emerge is whether different interaction patterns of individuals and their meaningful interpretation can be captured for social network analysis. In this work, we present a novel solution that discovers occurrences of prototypical 'ego network' patterns from social media and mobile-phone networks, to provide a data-driven instrument to be used in behavioral sciences for graph interpretations. We analyze nine datasets gathered from social media websites and mobile phones, together with 13 network measures, and three unsupervised clustering algorithms. Further, we use an unsupervised feature similarity technique to reduce redundancy and extract compact features from the data. The reduced feature subsets are then used to discover ego patterns using various clustering techniques. By cluster analysis, we discover that eight distinct ego neighborhood patterns or ego graphs have emerged. This categorization allows concise analysis of users' data as they change over time. We provide fine-grained analysis for the validity and quality of clustering results. We perform clustering verification based on the following three intuitions: i) analyzing the clustering patterns for the same set of users crawled from three social media networks, ii) associating metadata information with the clusters and evaluating their performance on real networks, iii) studying selected participants over an extended period to analyze their behavior.

Introduction

Extracting effective features for nodes of a given graph is a pivotal step for many graph mining tasks, such as mining across different graphs from the same domain, identifying leader nodes in graphs or outliers detection. One type of method examines the graph structure around individual users, so-called *ego graphs*. There are various studies conducted to analyze different properties of ego networks in social media and spatial-temporal networks (Wehrli 2008; Chittaranjan, Blom, and Gatica-perez 2011; Staiano et al. 2012; Pan, Aharony, and Pentland 2011; McAuley and Leskovec 2012; Arnaboldi et al. 2012). However, despite that, there are many issues that are hardly addressed for ego graphs. One such challenging problem is detecting distinct neighborhood patterns from ego graphs. This is an interesting research problem and involves many challenging issues that need to be addressed, such as (dynamic) characteristics of the data, selection of appropriate features, choosing an appropriate clustering technique, extraction of knowledge from clusters and above all the effective interpretation of ego clusters. For practical usage, such frameworks can be used in psychological studies to monitor the mood swings of patients, identifying emerging leaders in disaster affected areas and analyzing communication patterns of people in epidemic, etc. Similarly, some other specific scenarios can be i) given such ego graphs for several months, can we detect a set of distinct neighborhood patterns within them and use this information to get important cues for the next days? Here, a key step relies on the ability to extract effective features from each node that would best capture its characteristics, so that we can cluster the nodes, ii) given spatial-temporal phone data of students living together in a dormitory, can we design a tool that will help identify their communication patterns over time, to identify events such as holidays, exams, or a flu epidemic?, iii) after extracting distinct neighborhood patterns from ego graphs, can we use this information to infer social behavior (personality traits) of the people? These are of course only some of the many cases.

Current Work. In this work, we propose a novel solution DUKE, **D**iscover **U**seful **K**nowledge from **E**go graphs, for graph mining tasks of data from social media and spatial-temporal networks. This work aims to provide a data-driven instrument for graph interpretations, to be used in behavioral sciences. Ego networks are explored to discover the occurrences of prototypical neighborhood patterns by categorizing prototypes of interactions between users, and to assess their changing role within the social networks. We use five social media and four spatial-temporal datasets for this, with 13 network level features and three unsupervised clustering algorithms to detect clusters from the data. In our experiments we use social media, namely Facebook, Twitter, Foursquare, Wikipedia, collaboration networks and spatial-temporal mobile-phone data, in particular from Bluetooth, GPS and call-logs. For feature extraction, we use four centrality features (degree, betweenness, closeness and eigenvector), three efficiency features (global, local and nodal), two transitivity features (global and local) and four actor-based features (ego density, ego neighbors, dominant edges

Copyright © 2015, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

and ego weight). Similarly, we use k-means, hierarchical and affinity propagation for clustering purposes. We utilize an unsupervised feature similarity technique to identify a set of compact and less redundant features from the data, concurrently reducing the likelihood for the inclusion of spurious features to exaggerate the clustering results. We provide an empirical validation for assessing the quality of the delivered clusters. Our analysis yields several insights that not only reveal unique patterns within the data, but also empirical evidence of a limit on the maximum possible neighborhood patterns. In our experimental setting, we are particularly interested in what useful knowledge can we discover from clustering results. We choose three possible scenarios for knowledge extraction: i) we analyze the clustering patterns for the same set of users from three social media networks. We extract a common knowledge for users to study how their behavior in different networks correlates to each other, ii) each node in a social network has certain attributes and characteristics that identify its position within a network. To identify the relationship between the node attributes and its characteristic patterns, we make use of multi-label predictor fed with the extracted clusters as input, with the aim of correctly classifying the metadata attached to the nodes in real life, iii) we study clustering patterns for a small sample over a long period of time to understand their behavior at different time instances. To the best of our knowledge, this is the first work that explores various graph measures on social networks to automatically infer distinct neighborhood patterns from ego networks.

The remainder of the paper is organized as follows: we first present the background and related work on ego networks, followed by a description of our methodology. We then discuss the datasets and experimental results followed by the conclusion of the paper.

Background and Related Work

In this section, we discuss some basic graph notions and summarize the related work.

Background

An undirected graph is denoted by $G(V, E)$ where V is a set of nodes and E is a set of edges. Given a graph G and a node $v \in V$, Ego Network (v, G) is a sub-graph $\hat{G}(\hat{V},\hat{E})$, where $\hat{V}$ represents the direct neighbors of v. Ego networks are subnetworks that are centered on a certain node. In this work, we explore the first and second order neighborhood of an ego network. Table 1 summarizes the used network features. The feature details are reported in our technical report (Muhammad and Van Laerhoven 2015).

Feature Category	Selected Features
Centrality Measures (Freeman 1978)	Degree^{1a}, Betweenness^{1b}, Closeness^{1c}, Eigenvector^{1d}
Efficiency Measures (Latora and Marchiori 2003)	Global^{2a}, Local^{2b}, Nodal^{2c}
Transitivity Measures	Global^{3a}, Local^{3b}
Actor Based Measures	Ego Density^{4a}, Ego Neighbors^{4b}, Dominant Ego Edges^{4c}, Ego Weight^{4d}

Table 1: Extracted network features for ego graphs. We will use the superscript notions in the later sections (See section 'Feature Selection from the Datasets' and particularly Table 3).

Related Work

In this section, we review research works closely related to ours, from two distinct fields: i) ego networks analysis using psychological and network theory measures and, ii) discovering the patterns using graph mining.

Ego Network Analysis Using Psychological and Network Theory Measures A considerable amount of attention is devoted on studying five-personality traits from survey, spatial-temporal and web mining data (Staiano et al. 2012; Chittaranjan, Blom, and Gatica-perez 2011; Wehrli 2008; Pan, Aharony, and Pentland 2011). Staiano et al. used network level features on a spatial-temporal dataset collected in an undergraduate student campus to investigate personality traits. Their research shows that Bluetooth data identified different personality traits much better than call-log, survey, Bluetooth and call-log data together. Chittaranjan, Blom, and Gatica-perez developed an automated system for classifying personality traits based on actor level features, such as the use of camera, youtube videos, incoming call duration, Bluetooth information, etc. Wehrli predicted personality traits from the social networking website StudiVz using network and actor based features. Pan, Aharony, and Pentland studied spatial-temporal and survey based data of ego networks to identify the existence of individual-level correlation between financial status and interaction patterns and their connection to personality traits.

Apart from the five-traits model, other models are also developed for studying ego networks (Stoica and Prieur 2009; McAuley and Leskovec 2012; Arnaboldi et al. 2012; Henderson et al. 2011). Stoica and Prieur presented a model to characterize undirected graphs by enumerating small induced sub-structures. McAuley and Leskovec developed a probabilistic model to infer social circles (friends, family, college friends) from social network data. Arnaboldi et al. analyzed twitter data to identify the social circles within the ego networks. Henderson et al. proposed a feature extraction model that recursively combines local (in and out degree, total degree) and neighborhood (number of within-egonet edges, and the number of edges entering and leaving the ego net) features to produce behavioral information. Ego networks have also been studied in the health-care domain (Madan et al. 2010; O'Malley et al. 2012). Madan et al. analyzed Bluetooth scans, SMS networks, self-reported diet and weight-related information collected periodically over a nine-months period. Malley et al. discussed relationships between different network features and how these properties can be studied together in a health-care domain.

Discovering the Patterns Using Graph Mining The problem of finding subgraph patterns has been addressed using apriori (Yan et al. 2008; Yan and Han 2002; Kuramochi

and Karypis 2001) and pattern growth methods (Tong et al. 2007; Zhu et al. 2011). Yan et al. developed a classifier that exploits the correlation between structural similarity and significance similarity in a way that the most significant patterns can be identified quickly by searching dissimilar graph patterns. Yan and Han built a lexicographic order among the graphs and then map each graph to a unique depth-first search code and its canonical label. Later on, based on the lexicographic order, it adopts the depth-first search to mine the frequent connected subgraphs. Kuramochi and Karypis used the idea of adjacent representation of graphs and an edge-growing strategy to discover the frequent subgraph patterns. Frequent graph pattern mining using apriori has some serious procedural problems. A frequent n-edge labeled graph may contain 2^n frequent subgraphs. The presence of too many patterns may not lead to much precise knowledge. Pattern growth methods concentrate on representing patterns that preserve most of the information. Tong et al. presented a SQL-based method for finding best-effort subgraph patterns in attributed graphs, i.e. graphs with some attribute value attached to it. Zhu et al. developed a pattern growth approach where a small set of high potential subgraphs is discovered and then large subgraphs are detected by combining many smaller structures. The authors in (Moustafa, Deshpande, and Getoor 2012) developed SQL-based declarative language queries, where a given predefined structural pattern is searched for in every node's neighborhood and the counts are reported. To the best of our knowledge, we could not find any graph mining technique to cluster ego graphs from data.

Methodology

We now discuss our methodology for extracting ego clusters.

Unsupervised feature selection. We use feature selection using feature similarity (FSFS) (Mitra, Murthy, and Pal 2002) to identify feature subsets from the data. Our selection of an unsupervised feature selection technique is motivated by the following three intuitions: i) looking for every permutation is time consuming and causes delay for huge datasets. Mostly, the ego networks are without any class label which makes it very difficult to manually find any suitable feature subsets, ii) similarly, it is inconceivable to identify a particular feature subset that produces promising results regardless of the particular characteristic of the data, iii) FSFS has low computational complexity with respect to the number of features and number of samples in the data.

Now, we briefly summarize the FSFS approach. FSFS uses the maximal information compression index defined as:

$$2\lambda_2 = vr(x) + vr(y) - \sqrt{(vr(x) + vr(y))^2 - 4vr(x)vr(y)(1 - (p(x,y))^2)}) \quad (1)$$

where $vr(x)$, $vr(y)$ and $p(x, y)$ denote the variance of feature x, y and correlation coefficient respectively. The lowest value of index (zero) suggests that the two features are linearly dependable on each other and increases as the amount of dependency decreases. FSFS works in two phases, namely partitioning the original feature set into homogeneous subsets and selecting a representative feature from each such cluster. In the partitioning phase, it computes the k nearest neighbors (features) of each feature using the measure discussed above. Among them the feature having the most compact subset is determined and its k neighbors are discarded. The process is iteratively repeated until all of them are either selected or rejected.

Feature evaluation. We use entropy (Mitra, Murthy, and Pal 2002) and representation entropy (Dash and Liu 2000) measures to evaluate the effectiveness of the feature subsets. We minimize the likelihood for the inclusion of any spurious feature. The entropy can be defined as:

$$E = \sum_{i=1}^{N}\sum_{j=1}^{N}(s_{ij}.log(s_{ij}) + (1 - s_{ij}).log(1 - s_{ij})), \quad (2a)$$

$$s_{ij} = e^{-\alpha.dist_{ij}}, \quad (2b)$$

$$\alpha = \frac{-log(0.5)}{\overline{dist}}, \quad (2c)$$

Here, $dist_{ij}$ and $\overline{dist}$ represent the Euclidean distance between data items i and j and the mean dissimilarity between items in the dataset for a given feature subspace. Similarly, a representational entropy can be defined as:

$$H_R = -\sum_{j=1}^{d}\hat{\lambda_j}log\hat{\lambda_j} \quad (3)$$

where $\hat{\lambda_j}$ represents the m x m covariance matrix of a feature set of size m. We expect that the final reduced feature sets have low redundancy, i.e. a high H_R.

Data normalization and dimensionality reduction. In our case, the input feature spaces are high-dimensional. The performance of most clustering algorithms tends to scale poorly on high dimensional data. For this reason, we select the principle component analysis (PCA) for the dimensionality reduction. We normalize the features using equation 4 prior to applying it for dimensionality reduction.

$$V^* = \frac{V - min(V)}{max(V) - min(V)} \quad (4)$$

In this equation, V denotes the variable that is normalized, min and max indicate the two extremes of the variable.

Detecting the optimal number of clusters from clustering algorithms. To perform clustering, we select three well-known standard unsupervised clustering algorithms: i) k-means, ii) hierarchical clustering, and iii) affinity propagation (AP). We select different clustering algorithms to find out the best clustering algorithm that detects the optimal distinctive clusters. To identify the optimal number of clusters, we select the L-method (Salvador and Chan 2003) and the gap statistic (Tibshirani, Walther, and Hastie 2000). This step aims at identifying the clusters that are well separated, while penalizing an increasing number of clusters. The L-method was chosen for the hierarchical clustering algorithm due to its efficiency and good performance. The gap statistic is chosen for k-means. AP does not require the number of clusters to be determined before running it.

Clustering evaluation. We evaluate the clustering results using average Silhouette width of each cluster. It describes how well an element is assigned to the cluster. It is calculated using the mean intra-cluster distance $a(i)$ for each i and $b(i)$ is the lowest average dissimilarity of i to any other cluster of which i is not a member. The Silhouette width is defined as:

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}} \quad (5)$$

The average $s(i)$ over all data of a cluster is a measure of how tightly grouped all the data in the cluster is. We also use Jaccard and Rand indices to evaluate the clustering results of the same set of users crawled from different domains. Similarly, we assess the performance of the delivered clusters using a label prediction task.

Datasets

We select nine publicly available spatial-temporal, collaboration and social networking datasets. We use Bluetooth, GPS and call-log data from spatial-temporal datasets. We build a call network, where participants act as nodes and the number of calls between two nodes as edge weights. Similarly, Bluetooth and GPS networks are built with participants as nodes and the count of social interactions as edge weights. Spatial-temporal datasets often contain noisy edges or so called 'familiar strangers'. There are several techniques to prune out irrelevant edges. Thresholding is one of the popular techniques, but it is a one-size-fits-all solution, i.e. an edge may be relevant even with a low weight. We use (Serrano, Bogu, and Vespignani 2009) to select the relevant edges from the data. Similarly, we model social media and collaboration networks as unweighted and undirected graphs. Below we briefly discuss the datasets.

The Nokia dataset (Kiukkonen et al. 2010) contains the spatial-temporal data of 36 participants gathered between October 2009 and September 2011 from the French region in Switzerland. The dataset contains a wide range of behavioral data, such as Bluetooth, WiFi, GPS, Accelerometer, etc. **The Friends and Family dataset** (Aharony et al. 2011) contains the data collected between October 2010 and March 2011 from 40 individuals living in a married graduate student residence. The collected data has the Bluetooth, SMS and voice call data. **The MIT's Social Evolution dataset** (Madan et al. 2012) contains the data gathered between October 2008 and May 2009 from 74 participants living in a dormitory. The dataset contains scanned Bluetooth devices, logged call records, and SMS messages. **The Orange dataset** (Blondel et al. 2012) contains the ego networks of 4,357 mobile users collected in Ivory Coast by French Telecom between December 2011 and April 2012. **Facebook, Twitter and Foursquare datasets (Coscia et al. 2014)** contain the social networks of the same set of users in these three social media websites. **Network of famous philosophers** (Ahn, Bagrow, and Lehmann 2010) contains famous philosophers and their philosophical influences, as recorded by users of the English-language Wikipedia. Besides links to other philosophers, used to build the network, we also have the metadata representing the philosophical concepts, philosophical schools of thought, and so on. **Arxiv HEP-TH (High Energy Physics - Theory) collaboration network** (Leskovec and Krevl 2014) contains collaborations between authors papers submitted to Physics - Theory category. If an author i co-authored a paper with author j, the graph contains an undirected edge from i to j. A general overview of these networks can be found in Table 2, where $|N|$ and $|E|$ represent the number of nodes and edges respectively and $\overline{k}$ is the average degree of the network.

Network	$\|\mathbf{N}\|$	$\|\mathbf{E}\|$	$\overline{\mathbf{k}}$
The Nokia	36	147	8.17
The Friends and Family	40	501	24.75
The Social Evolution	74	2,526	68.27
The Orange	4,357	25,9110	59.06
Facebook	2,081	5,618	3.29
Twitter	3,745	31,638	8.71
Foursquare	5,738	42,691	8.23
Philosophers	1,231	5,978	9.7
HEP-TH	9,877	25,998	5.26

Table 2: Basic statistics of the networks studied.

Clustering Results

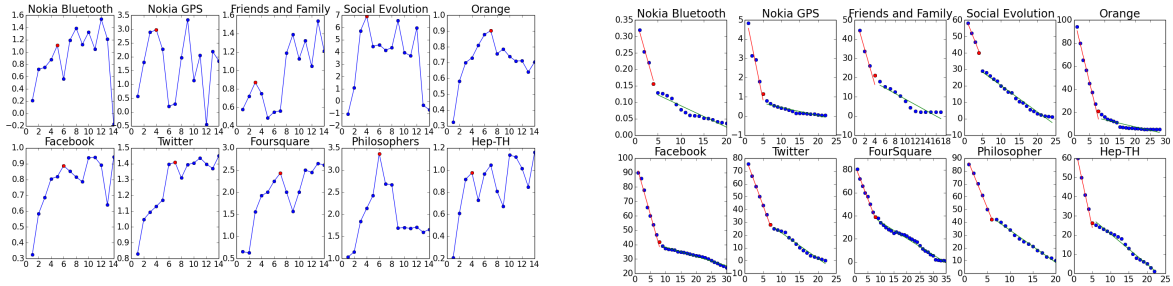
We now present our experimental results. We first concentrate on determining the distinct neighborhood structures. We evaluate the purity of the clusters. Next, we compare the clustering patterns for the same set of users across different networks. Since real world networks are enriched with annotated node information, we measure the ability of each cluster to predict the semantic information attached with the metadata of the nodes within the cluster itself.

Network	Selected Features	Entropy	$\mathbf{H_R}$
The Nokia Bluetooth dataset	$1^{(a,b,c,d)}, 2^b, 3^b$	0.31	1.99
The Nokia GPS dataset	$1^{(a,b,d)}, 4^{(a,d)}$	0.19	2.23
The Friends and Family dataset	$1^b, 2^b, 4^{(a,c)}$	0.25	1.31
The Social Evolution dataset	$1^b, 2^a, 3^b, 4^b$	0.10	0.80
The Orange dataset	$1^{(b,c)}, 2^b, 4^{(b,d)}$	0.45	7.30
The Facebook dataset	$1^b, 2^{(a,b)}, 3^a, 4^b$	0.68	2.38
The Twitter dataset	$1^{(b,d)}, 2^{(a,b)}, 3^a$	0.51	1.10
The Foursquare dataset	$1^{(a,b,c,d)}, \mathbf{2^a}, 3^{a,b}$	**0.39**	**1.22**
The Philosophers dataset	$1^b, 2^{(a,b)}, 3^{(a,b)}$	0.57	2.71
The HEP-TH dataset	$1^{(a,b,c,d)}, \mathbf{2^a, 3^a}$	**0.73**	1.70

Table 3: Feature selection by FSFS with entropy and representation entropy values. FSFS has detected different features for each dataset. See Table 1 for the notions. The bold letters show that the removal of these features from that particular combination improved the score.

Feature Selection from the Datasets

After extracting features from the raw data, we feed the extracted features to FSFS and record the reduced feature set. Table 3 shows the reduced feature subsets with their entropy and representation entropy scores. We notice that the reduced features vary for each dataset, i.e., different features are selected based on the particular characteristics of the



(a) Gap statistic for k-means

(b) L-method results for hierarchical clustering

Figure 1: The results of gap statistic and L-method for datasets. The red dots represent the optimum number of clusters.

data. It shows the significance of feature selection for unsupervised learning. We also perform exhaustive search on the features to verify the optimal feature subsets. We notice occasionally stronger results for some of the following eight combinations: (i) centrality measures; (ii) efficiency measures; (iii) transitivity measures; (iv) centrality and efficiency measures -i.e. the union of (i) and (ii); (v) centrality and transitivity measures -i.e. the union of (i) and (iii); (vi) efficiency and transitivity measures - i.e. the union of (ii) and (iii); (vii) four actor based features; $(viii)$ combination of all 13 measures. We select the features with minimum redundancy to avoid the inclusion of any spurious feature that exaggerates the clustering results. The entropy and H_R scores for the mentioned feature spaces are discussed in our technical report (Muhammad and Van Laerhoven 2015). In some cases, we notice that the selected eight feature subsets perform even better than FSFS. We especially report those cases. Overall, we notice that the reduced feature sets obtained from FSFS have the lowest entropy and highest representation entropy, except for the Hep-TH and the Foursquare dataset. For the Hep-TH dataset, the combination from FSFS has a slightly higher entropy, the entropy dropped from 0.73 to 0.65 for the combination of four centrality measures (feature set (i)). For the Foursquare dataset, the exclusion of global efficiency (2^a) decreased the entropy from 0.39 to 0.33 and H_R increased from 1.22 to 1.39 (feature set (v)).

Determining the Numbers of Clusters

For the given datasets, the applied gap statistic and L-method identify different possible clusters even for the same feature subsets. However, the optimal number of clusters identified for any combination of features are no more than eight. Figure 1(a) shows that k-means identifies a maximum of seven clusters from any dataset. The optimal numbers of clusters are detected for the Orange, Twitter and Foursquare datasets. It shows six clusters for the Facebook and Philosophers datasets. For smaller datasets, the maximum number of clusters is five. Figure 1(b) shows the results for the L-method. Overall, it identifies eight clusters for the Orange, Facebook and Foursquare datasets, seven clusters for the Twitter and six clusters for the Philosophers datasets. The lowest number of clusters is four for the Nokia Bluetooth, Friends and Family and Social Evolution datasets.

Evaluating Clustering Results

We choose the ideal number of clusters for the given datasets and then apply the respective clustering algorithm to detect the clusters. For each clustering result, we visually inspect the clustering patterns. We analyze the clustering patterns by again extracting its features. We find in total eight distinct ego graph patterns as shown in Figure 2. The prototypical clusters have the following properties and characteristics:

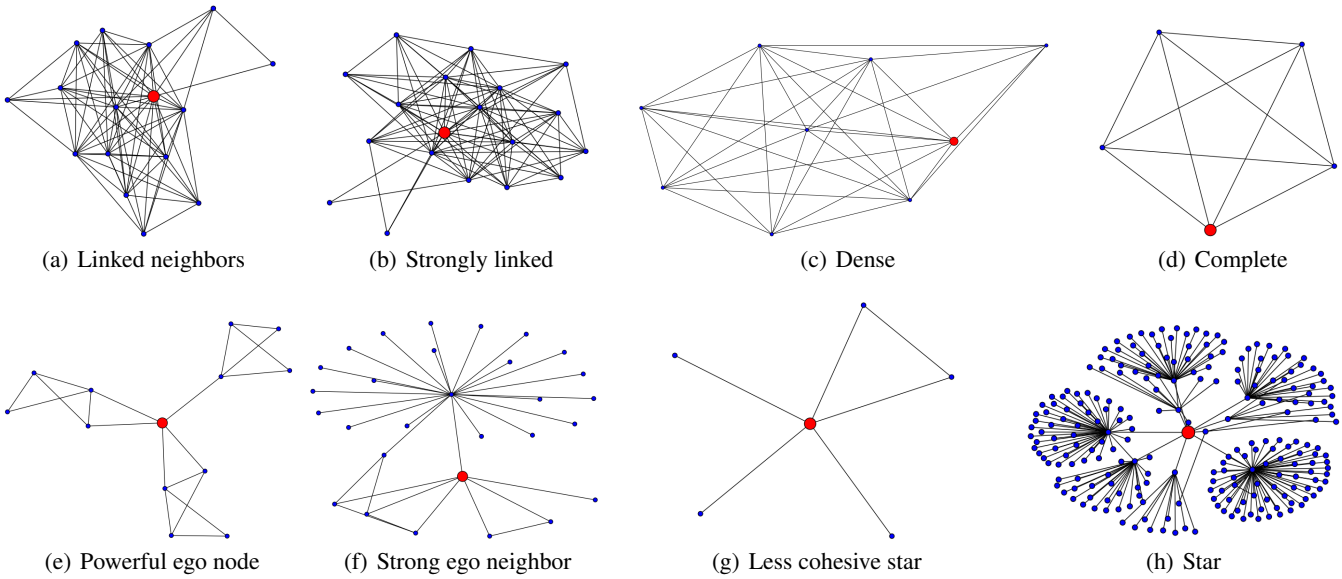
Cluster(a) (Linked neighbors): The ego node is the key player tied to active players. It is in a dense, active cluster at the center of events with many others. The ego has high closeness centrality and low degree- and betweenness centrality. The density of the graph is between 0.60 and 0.70. The ego has many immediate neighbors that are strongly connected to one another forming a strongly clustered network. The network has many complete structures (cliques) and the second order neighborhood is tightly connected.

Cluster(b) (Strongly linked): The ego has many immediate neighbors and high internal density. The ego has a highly populated second order neighborhood. The graph has many complete structures and the internal density is between 0.70 and 0.80. Even in case of removing the ego, there are multiple paths in the network to transfer the information.

Cluster(c) (Dense): The ego node is an active player in the network and contains a reasonable number of immediate neighbors. The neighbors of the ego node are well connected. The graph has high internal density between 0.80 and 0.90. Apart from very few neighbors (two or three), the remaining share a strong cohesion (bonding). Overall, the graph has many complete sub-graphs (cliques). Even in case of removing the ego from the network, it still contains many complete networks and the information can be easily transfered to other nodes. The ego's connections are highly redundant and most communication bypasses it.

Cluster(d) (Complete): The ego graph is complete and the density of the network is 1.0. It shows that the ego and its neighbors are actively in contact with each other.

Cluster(e) (Powerful ego node): The ego node is the most powerful player and removing it will paralyze the network. It has a high closeness and eigenvector centrality. The ego node acts as boundary spanners, i.e. it controls the communication between different parts of the network.



(a) Linked neighbors (b) Strongly linked (c) Dense (d) Complete

(e) Powerful ego node (f) Strong ego neighbor (g) Less cohesive star (h) Star

Figure 2: Clustering results of ego graphs, depicting the ego node in the middle (red color), with connections to first and second-degree neighbors. This study focuses on the automatic categorization of such ego graphs according to their graph structure. All in all, there exist eight distinct trends. We label each graph according to its characteristics.

Cluster(f) (Strong ego neighbors): The ego node has few immediate neighbors and the network is not densely connected. However, some nodes in the graph are highly populated and more powerful than the ego. The ego has a high closeness and degree centrality, but low betweenness centrality. The internal density is between 0.50 and 0.60.

Cluster(g) (Less cohesive star): The graph contains few small structures. It contains few trivial triads and the remaining neighbors form a star-shape network.

Cluster(h) (Star): Overall, the network has a sparse structure. The ego graph has a star structure and immediate neighbors of the ego are not connected. The number of neighbors varies depending upon the size of the network; the structure is small for smaller networks and large for bigger networks. The overall density of the ego graph is very low with no complete sub-graphs. Normally, such clusters are identified by a high value of centrality measures. In such cases, we found that the second order neighborhood of the ego is even more powerful and denser in terms of structure.

Table 4 represents the clustering results. We detect misclassified clusters using two sources: i) we select clusters with very low or negative Silhouette width as misclassified instances, ii) we rely on visual inspection of the small clusters. Similarly, we model Bluetooth and GPS modality from the Nokia dataset. We further categorize the clusters in four classes: i) linked structures ($C(a)$, $C(b)$), ii) dense structures ($C(c)$, $C(d)$), iii) informative ego ($C(e)$, $C(f)$), and iv) less dense ($C(g)$, $C(h)$). One shall note that the interpretation of the structures depends on many factors, such as the environment of data collection, context data and so on. In the later sections, we analyze some networks based on context information to give meaningful interpretation to the clusters. Now, we discuss major trends from Table 4.

Network	C(a)	C(b)	C(c)	C(d)	C(e)	C(f)	C(g)	C(h)	M
Nokia BT	+*!	+*!	+!	+*!			+*		
Nokia GPS	+*!	+*!	+*!	+*!			*		
Fri& Fam	+*!	*		+*!			+*!		
Social	+*!	+*!	+*!	+*!					
Orange	+*	+*	+*	*		+*	+*	+*	+*!
Facebook	+*	+*	+*	+*		*	+*	*	+*!
Twitter	+*	+*	+*	+*			+*	+*	+*!
Foursquare	+*	+*	+*	+*		*	+*	+*	+*!
Philosophers	+*			+*	+*	+*	+*	+*	!
Hep-TH				*	+*		+*	+*	+*!

Table 4: Clustering results for nine datasets using three algorithms. The first row represents the datasets. It contains the results for the features selected by FSFS. Similarly, $C(a)$,..., $C(h)$ represent the short form of cluster (a),..., cluster (h). The letter M represents the misclassified clusters in the datasets. We use three signs to represent clustering algorithms. The +, * and ! signs represent the k-means, hierarchical clustering and affinity propagation (AP) respectively. There are some cells without entries representing that for some combinations that particular shape is not detected.

Spatial-temporal clustering trends. Overall, we notice that extraction of possible clusters is largely dependent on the sample size and environment of data collection. The Social Evolution and the Friends & Family datasets are collected in a certain environment (student dormitory and married graduate students living in a campus facility) with people well familiarized with each other. Their clustering results illustrate that only certain clustering patterns are prominent. For the Social Evolution dataset, people are mostly confined within $C(a)$, $C(b)$, $C(b)$ and $C(d)$ that represent strongly clustered structures. Similarly, the Friends & Family dataset analysis shows that $C(a)$, $C(d)$ and $C(g)$ are prominent. The remaining patterns hardly exist in the data. The Orange dataset is gathered from a large sample of people liv-

	C(a)		C(b)		C(c)		C(d)		C(g)	
	S	I	S	I	S	I	S	I	S	I
Friends & Family	0.47	21					0.86	9	0.54	10
Social Evolution	**0.90**	20	**0.95**	30	**0.67**	7	**0.58**	17		

Table 5: Clusters– descriptive statistics of affinity propagation for the Friends and Family and the Social Evolution datasets. The Silhouette index (S) and number of instances (I) in each clusters.

ing in a diverse environment who hardly know each other. Their results show a large diversity of all possible clustering patterns. Similarly, we identify $C(a)$, $C(b)$, $C(c)$, $C(d)$ and $C(g)$ as prominent clusters for the Nokia dataset. Last, we briefly note that $C(a)$, $C(b)$, $C(c)$, $C(d)$ and $C(g)$ are prominent for the spatial-temporal datasets.

Social media clustering trends. The social media networks are relatively bigger in size than spatial-temporal datasets. Mostly, we observe similar patterns concerning their characteristics and qualitative properties; however, the frequency of their occurrences are unlike for the spatial-temporal datasets. We discover an interesting pattern $C(e)$ in the social media networks, especially in the Philosophers and the Hep-TH datasets. For example, we detect five prominent patterns for the Hep-TH dataset using hierarchical clustering, but the most prominent pattern is $C(e)$ that hardly existed in the spatial-temporal datasets. We notice in the Hep-TH dataset that the first authors act as a bridge between different disciplines, i.e. their research connects different branches. Similarly, the spatial-temporal datasets suffer from familiar stranger problem, which prevents certain shapes from emerging. We notice that $C(f)$ was more often visible in social media datasets than in spatial-temporal datasets. Last, we notice that $C(a)$, $C(b)$, $C(d)$, $C(g)$ and $C(h)$ are prominent clusters for social media datasets.

Quantitative comparison of clustering algorithms. Overall, when comparing the performance of the clustering algorithms, we notice that hierarchical clustering produces better clustering results concerning the purity of their existence inside clusters and margin of error. Similarly, we also note that all three techniques perform equally well on smaller datasets, but their performances deteriorate more or less for bigger datasets. Affinity propagation performs decent on smaller datasets, but generates many outliers for bigger datasets. We notice that it produces a large number of clusters for bigger datasets; especially for the Orange dataset, we observe that the number of clusters equals the sample size. Similarly, it detects more than 2,000 clusters for the Hep-TH dataset. We notice that the delivered clusters for bigger datasets are meaningless and incoherent. Last, we observe strong results for affinity propagation on the Social Evolution and the Friends and Family datasets as shown in Table 5. Similarly, the clustering results from k-means contain many misclassified instances for bigger datasets. Basically, k-means is stochastic in its initial conditions and therefore presents different trends for a fixed number of clusters for every run. Compared to k-means, we notice very few misclassified clusters for hierarchical clustering. Table 6 shows the clustering results for k-means and hierarchical clustering for four datasets along with their misclassified instances. We focus on the bigger datasets to investigate the purity of these patterns within the identified clusters and identify misclassified instances. We notice that k-means produces strong results for the Facebook and the Hep-TH datasets, where the clusters have high Silhouette width and fewer misclassified instances; especially only 14 misclassified instances for the Hep-TH dataset. We observe this cluster by visual inspection. Similarly, we notice only 84 misclassified instances for the Facebook dataset. However, we note that k-means detects a huge misclassified cluster for the Twitter and the Foursquare datasets. The misclassified clusters contain 368 and 597 instances with the Silhouette width of 0.05 and -0.25 respectively. For the Twitter dataset, Table 6 shows that the Silhouette width is very small for the clusters, which means the identified clusters are not tightly connected to each other. For hierarchical clustering, we identify best results for the Facebook, the Twitter and the Hep-TH datasets. There are some instances reported in Table 6 where hierarchical clustering identifies misclassified instances but the misclassified clusters are relatively smaller in size. We identify only 20 misclassified instances (a small cluster) for the Facebook dataset, which are relatively fewer as compared to 84 for k-means. However, for the Hep-TH dataset, k-means has only 14 misclassified instances as compared to 54 from hierarchical clustering, moreover, the Silhouette score is also relatively stronger for k-means. We also notice a misclassified cluster for the Foursquare dataset, it contains 254 instances and the Silhouette width is -0.10.

Clustering Evaluation across Social Networks

We now analyze the clustering results for the same set of users in different social networks. We analyze how clustering patterns for users correlate across different networks, i.e., do people follow the same communication patterns in social media and spatial-temporal networks. We use the Facebook, Twitter, Foursquare, Nokia and Orange datasets. We have the relationships of the same set of users in three social media websites. Similarly, we analyze the ego patterns from Bluetooth and GPS data to understand how much they correlate to each other. Last, for the Orange dataset, we use clustering patterns extracted at two different time instances. Further, we form different pairs of the networks. We make three pairs from social media (Facebook/Twitter, Facebook/Foursquare, Twitter/Foursquare), Bluetooth/GPS from the Nokia and T_1/T_2 from the Orange datasets. For each pair, we use the detected clusters as an input for the partition evaluation measures. We use the Jaccard and Rand metrics. Table 7 shows the results for the social media and spatial-temporal datasets. Overall, we notice that social media datasets achieve better results than spatial-temporal datasets. We obtain high scores for the Nokia dataset, but these values are still comparatively low considering the small sample size. Similarly, for the Orange dataset, we obtain maximum Jaccard and Rand similarity of 0.25 and 0.29 for hierarchical clustering; these scores are much lower for the remaining algorithms. We emphasize this for two reasons: i) Bluetooth and GPS data suffer from noisy edges, we prune them, but they still exist, ii) humans are dynamic in their activities, i.e., a person might call many people one

		C(a)		C(b)		C(c)		C(d)		C(e)		C(f)		C(g)		C(h)		M
		S	I	S	I	S	I	S	I	S	I	S	I	S	I	S	I	
k-means	Facebook	0.53	999	0.47	453	0.34	82	0.75	80					0.33	383			**84**
	Twitter	0.60	895	**0.17**	772	**0.24**	136	**0.19**	350					**0.16**	716	0.49	508	**368**
	Foursquare	0.53	1609	0.65	959	0.40	322	0.47	457					0.41	1128	0.50	666	**597**
	Hep-TH									0.62	5341			0.75	1023	0.70	3499	**14**
Hierarchical	Facebook	0.61	1090	0.51	399	0.75	46	0.59	115			0.87	4	0.74	320	0.93	114	**20**
	Twitter	0.58	1727	0.56	646	0.70	97	0.24	415					0.72	560	0.69	245	**55**
	Foursquare	0.61	2450	0.68	1087	0.79	181	0.35	420			0.53	189	0.67	707	0.71	425	**279**
	Hep-TH							0.39	783	0.55	3941			0.59	1940	0.61	3089	**54**

Table 6: Clusters– descriptive statistics for k-means and hierarchical clustering for four datasets. The S and I represent the Silhouette width and number of instances in a cluster, and M represents the number of instances in misclassified clusters.

	Jaccard Index					Rand Index				
	Social Media			Nokia	Orange	Social Media			Nokia	Orange
	F/T	F/FR	T/FR	Bluetooth/GPS	T_1/T_2	F/T	F/FR	T/FR	Bluetooth/GPS	T_1/T_2
k-means	0.34*	0.37*	0.25*	0.64	0.17*	0.39*	**0.43***	0.30*	0.71	0.21*
Hierarchical Clustering	0.42	0.43	0.33	0.62	**0.25**	0.47	**0.49**	0.38	0.68	**0.29**
Affinity Propagation	0.12*	0.14*	0.08*	0.57	0.11*	0.14*	0.17*	0.10*	0.61	0.13*

Table 7: Evaluation metrics for clustering algorithms. The F, T and FR denote Facebook, Twitter and Foursquare. Symbol * shows that the hierarchical clustering outperforms its competitors by 95% significance interval.

week and no one in the second week. Contrary, we obtain stronger results for the social media datasets with hierarchical clustering. The metrics scores for hierarchical clustering are relatively strong, which indicates that many users do follow similar communication patterns across social networks. The performance of affinity propagation deteriorates on the bigger dataset. Similarly, the pair of Facebook/Foursquare gives the best results with Rand index of 0.43 and 0.49 for k-means and hierarchical clustering. We notice an interesting trend for the Twitter/Foursquare pair as the metrics values are relatively lower (compared to the other two combinations) for all the algorithms. The misclassified clusters discussed in the previous subsection play a role in distorting the pairing results. Last, we also measure the statistical significance of performance differences between algorithms. The * symbols in Table 7 indicate a 95% significance interval. We notice that hierarchical clustering outperforms k-means and affinity propagation in all cases except for the Nokia dataset, where k-means delivers slightly better results.

Quality Evaluation via Label Prediction

We now turn to assess the quality of the delivered clusters using a label prediction task. We have two datasets, namely the Philosophers and the Friends and Family datasets, that have qualitative attributes of the nodes attached to them. We select nine unique attributes for each node from the Philosophers network, namely meta-physician, theologian, historian, political philosopher, physicist, analytic philosopher, socialist, mathematician and biologist. Similarly, the Friends and Family dataset contains a survey conducted on participants, which provides self-reported information about personality (Big Five). The participants were asked to use 1-5 point scales to answer the 44 questions Big Five questionnaire developed by (John and Srivastava 1999). The questionnaire owes its name to the personality traits shown in Table 8. The scores of the five traits are computed by summing the (inverted when needed) raw scores of the questions pertaining to each trait. The results mean and standard deviation are reproduced in Table 8. We perform the Kolmogorov-Smirnov goodness-of-fit test of normality on each data's distribution. All traits are normally distributed ($p < 0.05$). For each participant, we take the average score for each trait in range 0 and 1. Further, we assign one to the top three traits of each participant and zero to the remaining two. Afterwards, we attach the clustering membership of each node as known attributes, then its qualitative attributes as target labels to be predicted; we then feed this to a state-of-the-art binary classifier and record its performance.

Traits	Description	Mean	SD.Dev.
Agreeableness	assertive, friendly, cooperative	30.28	4.88
Conscientiousness	disciplined, organized, responsible	28.50	5.40
Extraversion	active, sociable, enthusiastic	23.13	6.71
Neuroticism	calm, unemotional	19.32	5.79
Openness	imaginative, intelligent, insightful	33.71	7.31

Table 8: Different traits along with their key descriptions.

We use a multi-label classifier (Tsoumakas and Katakis 2007), i.e. a classifier that assigns to each sample a set of target labels and is capable of predicting multiple target labels. We choose the binary relevance method (BR) because of its low computational complexity compared with other multi-label methods. The BR is a problem transformation strategy that decomposes a multi-label classification problem into L distinct single-label binary classification problems, $H_l : X \rightarrow \{l, \neg l\}$. Each binary classifier is then responsible for predicting the association of a single label. Hence this method trains $|L|$ binary classifiers $H_1, ..., H_L$

	Philosophers				Friends and Family			
	P	R	F1	H.loss	P	R	F1	H.loss
k-means	0.25	0.26	0.25	0.73	0.69	0.67	**0.68**	0.30
Hierarchical	0.34	0.37	**0.35**	**0.61**	0.73	0.72	**0.73**	0.26
AP	0.11	0.13	**0.12**	**0.86**	0.51	0.57	**0.54**	0.41

Table 9: Precision (P), Recall (R), F1 score (F1) and hamming loss (H.loss) for the Philosophers and the Friends and Family datasets.

by transforming the original dataset into L datasets $|D_l|$ that contain all examples of the original dataset, labeled as l if the labels of the original example contained l and as $\neg l$ otherwise. Each classifier H_j is responsible for predicting the $0/1$ association for each corresponding label $l_j \in L$. We provide the assigned cluster along with its attributes as an input to the classifier. Our datasets are of moderate size, so we feed all clustering results to the classifier. We use a multi-label version of Hamming loss, Precision and Recall for the label prediction. We further derive F1 score from Precision and Recall. Table 9 reports the results for the clustering algorithms on two datasets. Overall, hierarchical clustering outperforms its competitors. However, all three algorithms have strong performance on the Friends and Family dataset. The strong F1 scores hint that the detected clusters and the personality traits are correlated to each other. Similarly, we obtain a F1 score of 0.35 for hierarchical clustering on the Philosophers dataset, which is also a good score. However, it has a high Hamming loss of 0.61. The performance of AP greatly deteriorates for the Philosophers dataset with F1 score and Hamming loss of 0.12 and 0.86 respectively.

Case Study of Discovered Clusters

In this section, we present a case study of using the clusters extracted for the previous exposed evaluation. We demonstrate that extracted clusters have practical applications in the extraction of knowledge from real world scenarios. The results are derived from hierarchical clustering, because it mostly outperformed its competitors as discussed in the previous sections. We discuss the patterns for a sample of two participants reporting health issues from the Social Evolution dataset. The dataset contains results derived from a baseline questionnaire. The symptom survey was conducted for four months and contains questions regarding common contagious conditions – sore throat, runny nose, fever, nausea and mental health (depression and stress). Figure 3 represents the clustering results for two participants that reported continuous health issues for the survey period. The first plot represents a user often complaining about fever, nausea and mental health issues, the second plot represents a user complaining about a sore throat and a runny nose. We notice that for the first participant $C(g)$ and occasionally $C(h)$ (representing less dense clusters) are prominent patterns. This reflects the typical behavior of the people with these kinds of health issues: depressed and physically weak (fever/nausea) people tend to rest and communicate less, whereas people with less severe issues like a runny nose usually do not reduce communication with people. We notice that $C(a)$, $C(b)$ and $C(d)$ (all dense clusters) are prominent for the second user.

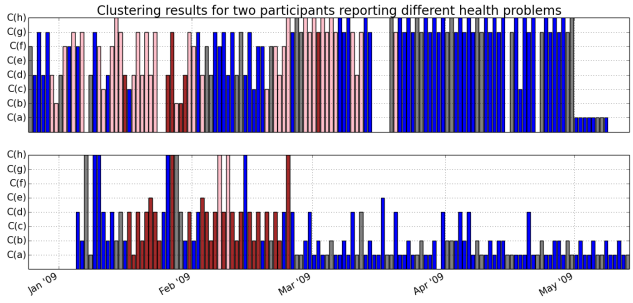


Figure 3: Clustering patterns for two participants reporting health issues. The blue and gray colors represent the participants and weekends respectively. Similarly, brown and pink colors represent two contagious diseases (sore throat, runny nose), and four other diseases (fever, nausea or mental health sickness) respectively.

Conclusion

In this work, we studied so-called ego graphs extracted from social media and spatial-temporal datasets to characterize their neighborhood patterns. To the best of our knowledge, this is the first study that focused on identifying neighborhood patterns in ego networks from such diverse domains. The major contributions of the paper are fourfold: i) we examined in a systematic way a wide range of network features (especially those addressing the characteristics of ego networks) and unsupervised techniques to identify the prototypical ego network patterns. Our results have shown that determining the optimal number of neighborhood patterns is surprisingly intricate. In addition, in case of a bad separation between the clusters, clustering algorithms tend to produce outliers and redundant clusters that can be misleading, ii) our clustering analysis detected eight prototypical emerging clusters for ego networks, each of them characterized by particular properties. We assigned labels to these prototypical clusters based on their shapes and properties of the ego and its neighborhood. We further categorized them into linked structures, dense structures, informative ego and less dense structures, iii) we investigated the purity of the delivered clusters within the identified clusters. Our analysis has shown that hierarchical clustering has produced optimum clustering results with minimum error margin. Further, we compared the clustering patterns across different social media and spatial-temporal networks to discover a common knowledge. Our experiments showed better results for the social media datasets, iv) we showed in our results section that this solution allows a discovery of useful patterns in different real world networks collected from information rich datasets.

Limitations and future directions. Now we discuss possible limitations and future directions of this research: i) Despite the fact that we analyzed a large collection of datasets, we are still not able to conclude an upper bound for the maximum possible number of ego patterns. It is possible that new shapes might emerge from large datasets, ii) We used standard graph theory measures for our analysis, but a new set of features will definitely shed more light on

the clustering results and possibly new clustering shapes, iii) The selection of algorithms based on the characteristics of the dataset is still an open problem. We have shown in our analysis that different algorithms have performed better for different datasets. Additionally, the traditional clustering algorithms are time consuming and not efficient for large datasets. They produce too many outliers for large dataset. A possible solution is to use BFR (Bradley, Fayyad, Reina) or $CURE$ (Clustering Using REpresentatives) for large datasets, iv) Another potential future direction would be to design a statistical model to efficiently infer clustering patterns with minimum error margin, v) We relied on visual inspection for smaller dataset, but this is not a feasible option for larger datasets. For that, the best options are the Silhouette width and inter-annotator agreement scores, vi) Last, the extraction of knowledge and interpretations of the clusters is merely dependent on many factors, such as environment and context of the data collection. A possible extension would be to collect data from different real-world scenarios and then interpret the clusters for specific scenarios.

References

Aharony, N.; Pan, W.; Ip, C.; Khayal, I.; and Pentland, A. 2011. Social fmri: Investigating and shaping social mechanisms in the real world. *Pervasive Mobile Computing*.

Ahn, Y.; Bagrow, J.; and Lehmann, S. 2010. Link communities reveal multiscale complexity in networks.

Arnaboldi, V.; Conti, M.; Passarella, A.; and Pezzoni, F. 2012. Analysis of ego network structure in online social networks. SocialCom.

Blondel, V.; Esch, M.; Chan, C.; F.;, C.; Deville, P.; Huens, E.; Morlot, F.; and Smoreda, Z.; Ziemlicki, Z. 2012. Data for development: the d4d challenge on mobile phone data.

Chittaranjan, G.; Blom, J.; and Gatica-perez, D. 2011. Who's who with big-five: Analyzing and classifying personality traits with smartphones. ISWC.

Coscia, M.; Rossetti, G.; Giannotti, F.; and Pedreschi, D. 2014. Uncovering hierarchical and overlapping communities with a local-first approach. *TKDD*.

Dash, M., and Liu, H. 2000. Feature selection for clustering. Springer-Verlag.

Freeman, L. 1978. Centrality in social networks conceptual clarification. *Social Networks*.

Henderson, K.; Gallagher, B.; Akoglu, L.; Li, L.; Eliassi-Rad, T.; Tong, H.; and Faloutsos, C. 2011. It's who you know: Graph mining using recursive structural features. KDD.

John, O., and Srivastava, S. 1999. The Big Five trait taxonomy: History, measurement, and theoretical perspectives. Handbook of Personality: Theory and Research.

Kiukkonen, N.; Blom, J.; Gatica-Perez, D.; and Laurila, J. 2010. Towards rich mobile phone datasets: Lausanne data collection campaign. ICPS.

Kuramochi, M., and Karypis, G. 2001. Frequent subgraph discovery. ICDM.

Latora, V., and Marchiori, M. 2003. Economic small-world behavior in weighted networks. *EPJ B*.

Leskovec, J., and Krevl, A. 2014. SNAP Datasets: Stanford large network dataset collection.

Madan, A.; Moturu, S. T.; Lazer, D.; and Pentland, A. 2010. Social sensing: Obesity, unhealthy eating and exercise in face-to-face networks. Wireless Health.

Madan, A.; Cebrin, M.; Moturu, S.; Farrahi, K.; and Pentland, A. 2012. Sensing the "health state" of a community. *IEEE Pervasive Computing*.

McAuley, J., and Leskovec, J. 2012. Discovering social circles in ego networks. NIPS.

Mitra, P.; Murthy, C.; and Pal, S. 2002. Unsupervised feature selection using feature similarity. *IEEE PAMI*.

Moustafa, W.; Deshpande, A.; and Getoor, L. 2012. Egocentric graph pattern census. ICDE.

Muhammad, S., and Van Laerhoven, K. 2015. Discovering neighborhood patterns in ego networks from mobile data. Technical report, Technische Universität Darmstadt.

O'Malley, A.; Arbesman, S.; Steiger, D.; Fowler, J.; and Christakis, N. 2012. Egocentric Social Network Structure, Health, and Pro-Social Behaviors in a National Panel Study of Americans. *PLoS ONE*.

Pan, W.; Aharony, N.; and Pentland, A. 2011. Fortune monitor or fortune teller: Understanding the connection between interaction patterns and financial status. SocialCom.

Salvador, S., and Chan, P. 2003. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. Technical report.

Serrano, M.; Bogu, M.; and Vespignani, A. 2009. Extracting the multiscale backbone of complex weighted networks. *NAS*.

Staiano, J.; Lepri, B.; Aharony, N.; Pianesi, F.; Sebe, N.; and Pentland, A. 2012. Friends don't lie: Inferring personality traits from social network structure. UbiComp.

Stoica, A., and Prieur, C. 2009. Structure of neighborhoods in a large social network. IEEE CSE.

Tibshirani, R.; Walther, G.; and Hastie, T. 2000. Estimating the number of clusters in a dataset via the gap statistic.

Tong, H.; Faloutsos, C.; Gallagher, B.; and Eliassi-Rad, T. 2007. Fast best-effort pattern matching in large attributed graphs. KDD.

Tsoumakas, G., and Katakis, I. 2007. Multi label classification: An overview. *IJDWM*.

Wehrli, S. 2008. Personality on Social Network Sites: An Application of the Five Factor Model. Working papers.

Yan, X., and Han, J. 2002. gspan: Graph-based substructure pattern mining. ICDM.

Yan, X.; Cheng, H.; Han, J.; and Yu, P. 2008. Mining significant graph patterns by leap search. SIGMOD.

Zhu, F.; Qu, Q.; Lo, D.; Yan, X.; Han, J.; and Yu, P. 2011. Mining top-k large structural patterns in a massive network. *PVLDB*.