Chapter 1

Uncovering criminal behavior with computational tools

Emilio Ferrara, Salvatore Catanese and Giacomo Fiumara

Abstract In this chapter we explore the opportunities brought in by advanced social network analysis techniques to study criminal behaviors and dynamics in heterogeneous communication media, along multiple dimensions including the temporal and spatial ones. To this aim, we present *LogViewer*, a Web framework we developed to allow network analysts to study combinations of geo-embedded and time-varying data sources like mobile phone networks and social graphs. We present some use-cases inspired by real-world criminal investigations where we used LogViewer to study criminal networks reconstructed from mobile phone and social interactions to identify criminal behaviors and uncover illicit activities.

1.1 Introduction

The pervasive diffusion of technologically-mediated communication channels pushed to unprecedented frontiers the ability of individuals to interconnect and exchange information. Mobile phone networks, social networking and media platforms like Facebook and Twitter, and over-IP messaging systems like Skype and WhatsUp,

Emilio Ferrara
School of Informatics and Computing, Indiana University Bloomington, 919 E. 10th St., Bloomington, IN 47408, USA, e-mail: ferrarae@indiana.edu

Salvatore Catanese
Department of Mathematics and Computer Science, University of Messina, viale F. Stagno D'Alcontres 31, I-98166 Messina, Italy
Department of Mathematics and Computer Science, University of Catania, viale Andrea Doria XX, Catania, Italy, e-mail: scatanese@unime.it

Giacomo Fiumara
Department of Mathematics and Computer Science, University of Messina, viale F. Stagno D'Alcontres 31, I-98166 Messina, Italy, e-mail: gfiumara@unime.it

represent some examples of the multitude of communication media broadly adopted in nowadays society. These phenomena generated lot of interest in the research community. Several aspects of socio-technical systems have been studied [63]: from macroscopic characteristics, like network structure [42, 37, 22, 21], to network dynamic, like information diffusion [54, 4, 47, 24], from microscopic behaviors, like how individual address their attention [39, 66] and what topics they discuss [18, 15], to social issues, like how people organize and mobilize using technology [31, 16, 17, 62] and what effects technological media have at the societal level [30, 43].

One aspect that has vast societal impact is the improper usage of such platforms. Technologies have been long exploited for criminal activities: for example, various studies showed how the Internet has been exploited for cybercrime, terror and militancy purposes [1, 12, 34]. In terms of abuse, mobile communication networks and social media have been mostly studied as vectors for the diffusion of computer viruses and malware [38, 33]. On the other hand, the possibility that such communication channels can be exploited by criminals to organize and coordinate their illicit activities in the physical world has been recently found very real [44, 45]. The ability to detect criminal behavior across different communication media is of paramount importance to avoid abuse and fight crime. For this reason, computational tools and models have been recently proposed to study criminal behavior in online platforms [69, 70, 71], social media [64], and mobile phone networks [23, 7]. Usually, such models and techniques are limited to one or few specific use-cases. For example, we recently proposed a tool called *LogAnalysis* that allows an investigator to reconstruct and visualize networks from mobile phone call data [13].

Here we present *LogViewer*, a next-generation Web-based criminal network analysis framework that yields advanced social network analysis functions, *de facto* extending *LogAnalysis* features to different types of networks, for example phone call networks and social graphs. *LogViewer* allows to study each network from three different angles: *(i)* static analysis, to investigate the role of nodes and edges, their centrality, and the emerging communities representing potential criminal rings; *(ii)* temporal analysis, to span across different temporal events and study the flow of information over time; finally, *(iii)* spatial analysis, embedding the network in a geographic space to determine physical closeness and locality effects on the network structure. *LogViewer* also allows to create multilayer spatio-temporal networks by merging different network types and to perform the above-mentioned different types of analysis on such a more complex network.

Our framework inherits different visualization layouts and algorithms from *LogAnalysis*: some of them are discussed in details in our previous work [13]. Here we first give an overview of the basic concepts borrowed by social network analysis and their meaning in criminal network analysis; this includes network centrality measure to identify roles in criminal networks, and community detection to unveil criminal gangs hidden within the network. After that, we present the new features provided by our criminal network analysis framework, especially *ad hoc* visualization meth-

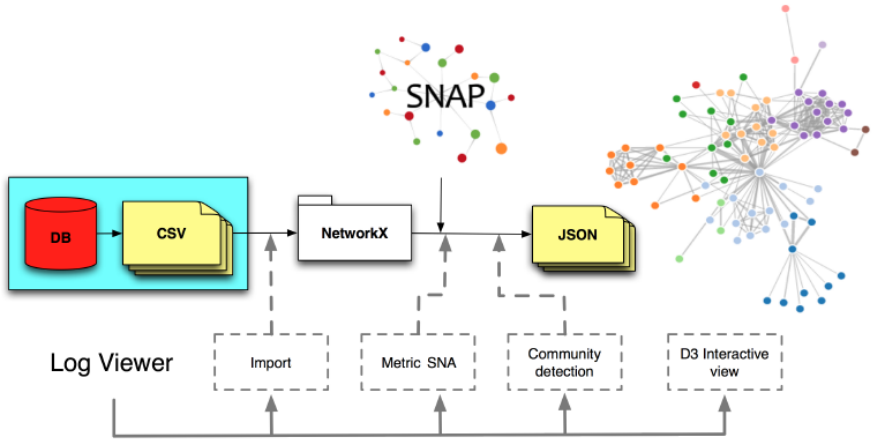


Fig. 1.1: Architecture of LogViewer.

ods that we devised keeping in mind the needs of law enforcement agencies, analysts and investigators. We illustrate these advanced criminal network analysis features by presenting examples or use cases inspired by real investigations, carried out by Italian law agencies, that benefited from the adoption of *LogViewer*.

1.2 *LogViewer* framework

1.2.1 Architecture and workflow

LogViewer is a Web-based framework that allows advanced network analysis on criminal networks reconstructed from various data sources, including (mobile) phone data and online social network data. It supports spatio-temporal analysis and it extends, *de facto*, the horizon of possibilities provided by *LogAnalysis* [13].

This framework implements various techniques of network generation, statistical measurement, partitioning (or clustering), and visualization that rely on powerful open-sources tools; the list includes GraphML for data storage, Python network libraries for data import, normalization and network representation like NetworkX[1] and iGraph, the Stanford Network Analysis Project (SNAP) library[2] to efficiently

[1] http://networkx.github.com/

[2] http://snap.stanford.edu/

compute network statistics, the Louvain method for network clustering [6], and the Javascript D3.js[3] library for interactive network visualization and exploration.

The architecture of LogViewer is represented in Figure 1.1. In the following we illustrate the typical workflow to bootstrap a criminal investigation using LogViewer. Let us use the example of data representing a mobile phone call network —the analysis of other sources, such as social network data, follows straightforwardly.

During an investigation, the agency in charge of it will obtain, usually through court warrants, raw data from a Telecommunication Service Provider related to the phone call interactions of a (possibly large) set of suspects involved in a certain criminal activity. Such data are generally provided in different formats: LogViewer allows some degree of standardization, supporting different formats adopted by various European service providers, e.g., Vodafone, Orange, and others.

The analyst can import one (or more) datasets into LogViewer, which will take care of appropriately reconstruct a network representation of such data, where each node corresponds to a given entity (generally speaking, in the mobile phone cases, the framework assumes a 1-to-1 mapping from phone to person, but it also supports the assignment of multiple phone numbers to the same entity, whereas such information is provided). Interconnections among entities, representing phone calls, are imported as links of this network. Duration and frequency of the calls are encoded in the network representation by means of different weighting systems that can be adopted by the analysts. For example, the raw number of calls between a pair of entities, or the average or total duration, among others, are available metrics that can be used for this purpose. This yields the possibility of performing dynamic network representation and temporal analysis.

In addition, each phone interaction reports geo-referenced data about the location of the caller and the called nodes (e.g., extrapolated from the GPS sensors on the mobile device, or approximated by the telephone cell corresponding to the physical location of the individuals at the time of the call); such information is attached to each event, to allow for spatial analysis. Once the data import procedure is completed, static representation (and spatio-temporal representation when meta-data are available) becomes available through LogViewer's visualization interface.

In the following, let us provide a bit more details about the type of data commonly processed by LogViewer for criminal network analysis purposes.

[3] http://d3js.org/

Table 1.1: An example of the structure of a phone log file.

Field	Description
IMEI	IMEI code MS
called	called user
calling	calling user
date/time start	date/time start calling (GMT)
date/time end	date/time end calling (GMT)
type	sms, mms, voice, data etc.
IMSI	calling or called SIM card
CGI	Lat. long. BTS company

1.2.2 Data and network representation

1.2.2.1 Mobile phone data

In the context of real-world investigations, mobile phone service providers, upon request by judiciary authorities, release data logs, normally in textual file format, with space or tab separation (CSV format). A typical log file contains, at least, the values shown in Table 1.1.

Similarly, information about owners of SIM cards, dealers of SIM cards and operations like activation, deactivation, number portability are provided by the service providers as additional material to ease and support the investigation activities. Log file formats produced by different companies are heterogeneous. *LogViewer*, first of all, parses these files and converts data into GraphML format. It is an XML valid and well-formed format, containing all nodes and weighted edges, each weight representing the various weighting strategies (e.g., the frequency of phone calls) used to represent the interactions between two connected nodes. GraphML has been adopted both because of its extensibility and ease of import from different SNA toolkits and graph drawing utilities.

1.2.2.2 Social graph data

Another rich source of information that is increasingly becoming adopted during criminal investigation is represented by Online Social Network data. Such types of datasets are provided by the Service Providers (like Facebook or Google) through court warrants to the law enforcement agency, similarly to mobile phone records.

Generally speaking, the datasets obtained by OSN service providers provide user meta-data related to the set of accounts of interest for the criminal investigation, including registration details (e.g., personal information, dates of account creation/deletion, etc.) along with the IP addresses corresponding to the devices used

for connection (and/or the GPS coordinates of the mobile device, in case any connection is performed in mobility). Logs include, among other data, the entire history of wall posts and comments, pictures and photographs, check-in events in specific physical locations, the chronology of incoming and outgoing friendship requests, the list of friends (on Facebook) or contacts (followers and followees on Twitter and similar platforms). Some platforms, like Facebook and Twitter, can provide detailed logs of personal interactions, such as chat or personal messages. Possibly, the same set of information is provided about any number of friends/contacts of the given individual target of the criminal investigation, if deemed relevant for the investigation by the judiciary authorities. Such data about the target's neighbors help enriching the amount of information available to LogViewer to perform its analysis.

LogViewer processes these datasets and extracts the information that can be put in form of network representation. For example, when reconstructing a social network, link weighting schemes represent the interactions (e.g., number of wall comments, frequency of chatting, etc.) between a pair of individuals. Although our framework does not yet provide advanced content analysis, such additional information is often adopted by the analysts by using external tools for traditional corpora analysis.

It's worth noting that, in the context of a criminal investigation, the analysts will study social network information with different lens, say in respect to the perspective of phone interactions. This is clearly due to the fact that online friendship, say on Facebook, has a very different meaning if compared to phone interactions. On the other hand, the possibility of performing further analysis on textual content produced by personal interactions (e.g., chat) eases the analysis, say with respect to phone calls monitoring and analysis (which might not be possible whereas recordings are not readily available for investigation purposes or need additional warrants to be accessed).

1.2.3 Data normalization and cleaning

Data clean-up usually means the deletion of redundant edges and nodes. This step is very importante since datasets often contain redundant information, that crowds graph visualization and biases statistical measures. In these circumstances, redundant edges between the same two nodes are collapsed and a coefficient – i.e., a edge weight – is attached, which expresses the number of calls. Our tool normalizes data after reading and parsing log files whichever format they have been provided among the standard formats (i.e., *fixed width text*, *delimited*, CSV, and more) used by mobile service providers.

1.3 Static analysis of criminal networks

1.3.1 Centrality measures

LogViewer takes into account the concept of *centrality measure* to highlight actors that cover relevant roles inside the analyzed network [46]. Several notions of centrality have been proposed during the latest years in the context of Social Network Analysis.

There are two fundamentally different class of centrality measures in communication networks. The first class of measures evaluates the centrality of each node/edge in a network and is called point centrality measure. The second type is called graph centrality measure because it assigns a centrality value to the whole network. These techniques are particularly suited to study phone traffic and criminal networks.

In detail, in *LogViewer* we adopted four point centrality measures (i.e., *degree*, *betweenness*, *closeness* and *eigenvector* centrality), to inspect the importance of each node of the network.

The set of measures provided in our tool is a selection of those provided by Social Network Analysis [65]. It could be not sufficient to solve any possible task in phone call network analysis. In fact, for particular assignments it could yet be necessary to use additional tools in support to *LogViewer* and in further evolutions we plan to incorporate new centrality measures.

For each centrality measure, the tool gives the possibility, to rank the nodes/edges of the network according to the chosen criterion. Moreover, *LogViewer* allows to select those nodes that are central, according to the specified ranking, highlighting them and putting into evidence their relationships, by exploiting the node-link layout techniques (discussed in the following). This approach makes it possible to focus the attention of the analysts on specific nodes of interest, putting into evidence their position and their role inside the network, with respect to the others.

In the following we formally describe the centrality measures used in *LogViewer*.

They represent the centrality as an indicator of the activity of the nodes (degree centrality), of the control on other nodes (betweenness centrality), of the proximity to other nodes (closeness centrality) and of the influence of a node (eigenvector centrality).

1.3.1.1 Degree centrality

The degree centrality of a node is defined as the number of edges adjacent to this node. For a directed graph $G = (V, E)$ with n nodes, we can define the in-degree and out-degree centrality measures as

$$C_D(v)_{in} = \frac{d_{in}(v)}{n-1}, \quad C_D(v)_{out} = \frac{d_{out}(v)}{n-1} \tag{1.1}$$

where $d_{in}(v)$ is the number of incoming edges adjacent to the node v, and $d_{out}(v)$ is the number of the outgoing ones.

Since a node can at most be adjacent to $n-1$ other nodes, $n-1$ is the normalization factor introduced to make the definition independent on the size of the network and to have $0 \leq C_D(v) \leq 1$.

In and out-degree centrality indicates how much activity is going on and the most active members. A node with a high degree can be seen as a hub, an active nodes and an important communication channel.

We chose to include the degree centrality for a number of reasons. First of all, is calculation is computationally even on large networks. Furthermore, in the context of phone call networks it could be interpreted as the chance of a node for catching any information traveling through the network.

Most importantly, in this type of directed networks, high values of in-degree are considered a reliable indicator of a form of popularity/importance of the given node in the network; on the contrary, high values of out-degree are interpreted as a form of gregariousness of the given actor in respect to the contacted individuals.

1.3.1.2 Betweenness centrality

The communication between two non-adjacent nodes might depend on the others, especially on those on the paths connecting the two nodes. These intermediate elements may wield strategic control and influence on many others.

The core issue of this centrality measure is that an actor is central if she lies along the shortest paths connecting other pairs of nodes. The betweenness centrality of a node v can be defined as

$$B_C(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{1.2}$$

where σ_{st} is the number of shortest paths from s to t and $\sigma_{st}(v)$ is the number of shortest paths from s to t that pass through a node v.

The importance of the betweenness centrality regards its capacity of identifying those nodes that vehiculate information among different groups of individuals.

In fact, since its definition due to Freeman [25] the betweenness centrality has been recognized as a good indicator to quantify the ability of an actor of the network to control the communication between other individuals and, specifically for this reason it has been included in *LogViewer*.

In addition, it has been exploited by Newman [48] to devise an algorithm to identify communities within a network. Its adoption in the phone traffic networks is crucial to identify those actors that allow the communication among different (possibly criminal) groups.

1.3.1.3 Closeness centrality

Another useful centrality measure that has been adopted in *LogViewer* is called *closeness centrality*. The idea is that an actor is central if she can quickly interact with all the others, not only with her first neighbors [49]. The notion of closeness is based on the concept of shortest paths (geodesic) $d(u,v)$, the minimum number of edges traversed to get from u to v. The closeness centrality of the node v is define as

$$C_C(v) = \frac{1}{\sum_{u \in V} d(u,v)} \tag{1.3}$$

Such a measure is meaningful for connected graphs only, assuming that $d(u,v)$ may be equal to a finite value.

In the context of criminal networks, this measure highlights entities with the minimum distance from the others, allowing them to pass on and receive communications more quickly than anyone else in the organization. For this reason, the adoption of the closeness centrality is crucial to put into evidence inside the network, those individuals that are closer to others (in terms of phone communications).

In addition, high values of closeness centrality in such type of communication networks are usually regarded as an indicator of the ability of the given actor to quickly spread information to all other actors of the network. For such a reason, the closeness centrality has been selected to be included in the set of centrality measures adopted by *LogViewer*.

1.3.1.4 Eigenvector centrality

Another way to assign the centrality to an actor of the network in *LogViewer* is based of the idea that if a node has many central neighbors, it should be central as well. This measure is called *eigenvector centrality* and establishes that the importance of a node is determined by the importance of its neighbors.

The eigenvector centrality of a given node v_i is

$$C_E(v_i) \propto \sum_{u \in N_i} A_{ij} C_E(u) \tag{1.4}$$

where N_i is the neighborhood of the given node v_i, and $x \propto Ax$ that implies $Ax = \lambda x$. The centrality corresponds to the top eigenvector of adjacency matrix A.

In the context of telecom networks, eigenvector centrality is usually regarded as the measure of influence of a given node. High values of eigenvector centrality are achieved by actors who are connected with high-scoring neighbors, which in turn, inherited such an influence from their high-scoring neighbors and so on.

This measure well reflects an intuitive important feature of communication networks that is the influence diffusion and for such a reason we decided to include the eigenvector centrality in *LogViewer*.

1.3.1.5 Clustering coefficient

The clustering (or transitivity) coefficient of a graph measures the degree of interconnectedness of a network or, in other words, the tendency of two nodes that are not adjacent but share an acquaintance, to get themselves in contact. High clustering coefficients mean the presence of a high number of triangles in the network.

The local clustering coefficient C_i for a node v_i is the number of links among the nodes within its neighborhood divided by the number of links that could possibly exist among them

$$C_i = \frac{|\{e_{jk}\}|}{k_i(k_i-1)} : v_j, v_k \in N_i, e_{jk} \in E \tag{1.5}$$

where the neighborhood N of a node v_i is defined as $N_i = \{v_j : e_{ij} \in E \wedge e_{ji} \in E\}$, while $k_i(k_i-1)$ is the number of links that could exist among the nodes within the neighborhood.

In is well-known in the literature [65] that communication networks show high values of clustering coefficient since they reflect the underlying social structure of contacts among friends/acquaintances. Moreover, high values of local clustering coef-

ficient are considered a reliable indicator of of nodes whose neighbors are very well connected and among which a substantial amount of information may flow. For such a reason, *LogViewer* provides the possibility of computing both the global clustering coefficient for any given phone call network and the local clustering coefficient of any given node.

1.3.2 Community detection in criminal networks

A criminal network can be regarded as a special kind of social network in which attention is devoted to secrecy and efficiency, since its members must communicate without being detected [67]. On the other hand, the crucial task of uncovering the functionalities of a criminal organization can be accomplished only by acquiring knwoledge about the structure of the underlying criminal network. Criminal networks usually exhibit diversified compositions: hierarchical [55], cellular [61] and flat structures [36] are the most common. One of the most relevant features of graphs representing real networks like criminal networks is the emergence of clustering phenomena, or communities. The detection of communities in criminal networks brings, as a main consequence, the identification of groups and their structures via the information coded in the topology of the corresponding graph.

The problem of finding communities in a network is often expressed as a clustering problem. A widely adopted approach to solve this problem is based on the concept of *network modularity* which can be expressed as follows: given a network, represented by means of a graph $G = (V, E)$, which has been partitioned into m communities, its corresponding value of network modularity is

$$Q = \sum_{s=1}^{m} \left[\frac{l_s}{|E|} - \left(\frac{d_s}{2|E|} \right)^2 \right] \tag{1.6}$$

assuming l_s the number of edges between vertices belonging to the s-th community and d_s is the sum of the degrees of the vertices in the s-th community. High values of Q imply high values of l_s for each discovered community, yielding to communities internally densely connected and weakly coupled among each other.

The network modularity is therefore used as fitness function to solve an optimization problem: among the several methods we mention here the Girvan and Newman (GN) algorithm [29], and an optimized variant known as Newman's algorithm [50], which is fast enough to support interactive real-time adjustments. *LogViewer* provides two strategies for detecting communities, namely the already cited Newman's algorithm and the Louvain method [6], another modularity maximization algorithm that performs very well with larger networks.

We recently discussed in great detail the problem of detecting communities and gangs inside criminal networks [23], and we point the reader's attention toward that work for an in-depth treatment of this topic.

1.3.3 Criminal network visualization

Typical network visualization tools rely on the popular force-directed layout [27]. The force-directed model represents the structure of the graph on the same foot as a physical system, in which nodes are physical points subject to various forces; nodes' coordinates (and therefore the layout itself) derive from the search of an equilibrium configuration of the physical system modeled by the algorithm [9]. This particular layout arrangement has the advantage of grouping users in clusters which can be identified according to the heightened connectivity. The Barnes-Hut algorithm [5] associated to this layout simulates a repulsive N-body system to continuously update the position of the elements.

To optimize the visualization, it is possible to interactively modify the parameters relative to the tension of the springs (edges). Nodes with low degree are associated a small tension and the elements are located in peripheral positions with respect to high degree nodes. Other parameters can be tuned, such as spring tension, gravitational force and viscosity. Our goal, in the following, is to suggest two methods to improve force-directed based layouts. As we will show, these techniques are especially well suited for criminal network analysis; however, they could potentially be generalized for broader usage in other domains of network analysis — for example, for applications in social and political sciences.

For the usage of traditional network visualization methods in criminal network analysis the reader should consult our recent paper on *LogAnalysis* [13].

1.3.3.1 Focus and context based visualization

The number of edges within a network usually grows faster than the number of nodes. As a consequence, the network layout would necessarily contain groups of nodes in which some local details would easily become unreadable because of density and overlap of the edges. As the size and complexity of the network grow, eventually nodes and edges become indistinguishable. This problem is known as visual overload [2]. A commonly used technique to work around visual overload consists of employing a zoom-in function able to enlarge the part of the graph of interest. The drawback of this operation is the detriment of the visualization of the global structure which, during the zooming, would not be displayed. However, such

a compromise is reasonable in a number of situations including, in some cases, the domain of criminal network analysis.

During an investigation, it is crucial to narrow down the analysis to the relevant suspects, to efficiently employ human and computational resources. Police officers typically draw some hypotheses about an individual suspect of being part of a criminal organization, or of being involved (or about to) in some crime; they concentrate the initial investigation on this individual, and on that person's social circles, as a ground to build the social network object of analysis. The main role of visual analysis lies in allowing the detection of unknown relations, on the base of the available limited information. A typical procedure starts from known entities, to analyze the relations with other subjects and continue to expand the network inspecting first the edges appearing the most between individuals apparently unrelated. During this procedure, only some nodes are relevant and it is important to focus on them rather than on the network as a whole.

Nevertheless, a spring embedded layout (including force-directed ones) does not provide any support to this kind of focus and analysis. In these situations, *focus and context* visualization techniques are needed to help a user to explore a specific part of a complex network. To this purpose, we here introduce the fisheye and the foci layouts.

1.3.3.2 Fisheye visualization

Focus and context is an interactive visualization technique [40]. It allows the user to focus on one or more areas of a social network, to dynamically tune the layout as a function of the focus, and to improve the visualization of the neighboring context. The *fisheye view* is a particular focus and context visualization technique which has been applied to visualize self-organizing maps in the Web surfing [72]. It was first proposed by Furnas [28] and successively enriched by Brown et al. [56]. It is known as a visualization technique that introduces distortion in the displayed information.

The fisheye layout is a local linear enlargement technique that, without modifying the size of the visualization canvas, allows to enhance the region surrounding the focus, while compressing the remote neighboring regions. The overall structure of the network is nevertheless maintained. An example of application of this technique is show in Figure 1.2. The picture shows a moderately small criminal network reconstructed from phone call interactions of about 75 individuals. The layout on the left panel is obtained by using a force-directed method implemented in our framework, *LogViewer*. The analyst can inspect the nodes of the network, which contains known criminals, suspects, and their social circles. When the focus is applied on a given node, the visualization transitions to the fisheye layout (see the right panel). A tool-tip with additional information about the node appears when the node is selected — it shows the phone number, personal details, address, photo, etc. The

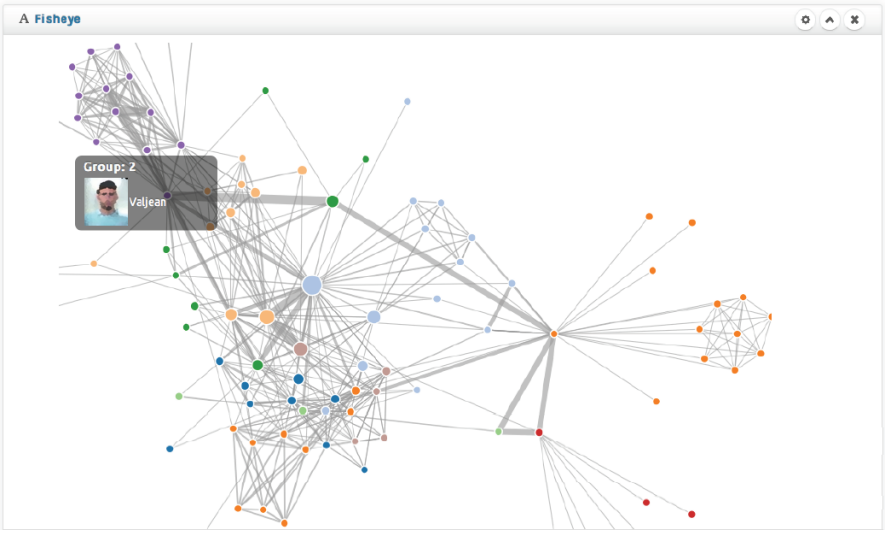


Fig. 1.2: Fisheye visualization.

layout causes edges among remote nodes to experience stronger distortions than local nodes. The upside of the presented method is the possibility to achieve the three recommendations of Network Nirvana [57] when focusing on a given node: all the nodes' neighbors are clearly visible, the node degree is easily countable, and the edges incident on that node can be identified and followed.

Note that fisheye and force-directed layouts can be used in conjunction. By combining the two methods, our framework efficiently yields focus and context views.

1.3.3.3 Matrix layout

A network can be represented by using an adjacency matrix in which each cell ij represents the edge existing between the vertex i and the vertex j. In our case, the vertices represent the phone numbers of the users (the caller and the called), and the edges represent their contacts. The natural visualization technique associated to this two-dimensional representation of the graph is the matrix layout. Nevertheless, the efficiency of a matrix diagram strongly depends upon the order of rows and columns: if the nodes that are connected are placed in order, then clusters and connections among communities can be easily identified. As shown in Figure 1.3, matrix cells can be coded to show additional information: in this case different colors represent different clusters.

Fig. 1.3: Matrix layout and clustering.

On the contrary of node-link diagrams, matrix layout makes not easily identifiable the paths connecting the vertices. On the other hand, when dealing with highly connected networks, the node-link layout rapidly becomes unreadable as a consequence of the large superposition of nodes and edges.

1.3.3.4 Foci layout

The *foci layout* implements three network visualization models: force-directed, semantic and clustered layouts. The latter is based on the Louvain community detection algorithm [6]. Future implementations will explore other methods [19, 20]. Our model supports multilayer analysis of the network through interactive transitions from the force-directed layout, with a single gravitational center, to the clustered one with more force centers placed in predetermined distinct areas. This layout allows to analyze the network on various layering levels depending on specified node attributes. Figure 1.4 shows the phone traffic network of some clans the previous criminal network, in which the color of the nodes denotes the type of crime committed by the members.

In this example, the clustering truthfully reflects the known territorial division among the groups belonging to the organization. In Figure 1.4 the focus is on a specific node. Using this layout it is possible to contextually analyze the community structure, the type of committed crime in respect to the members of the clan, and the direct relations of each single individual. This layout integrates also the forth Network Nirvana recommendation, namely the possibility to identify clusters and to highlight the community structure.

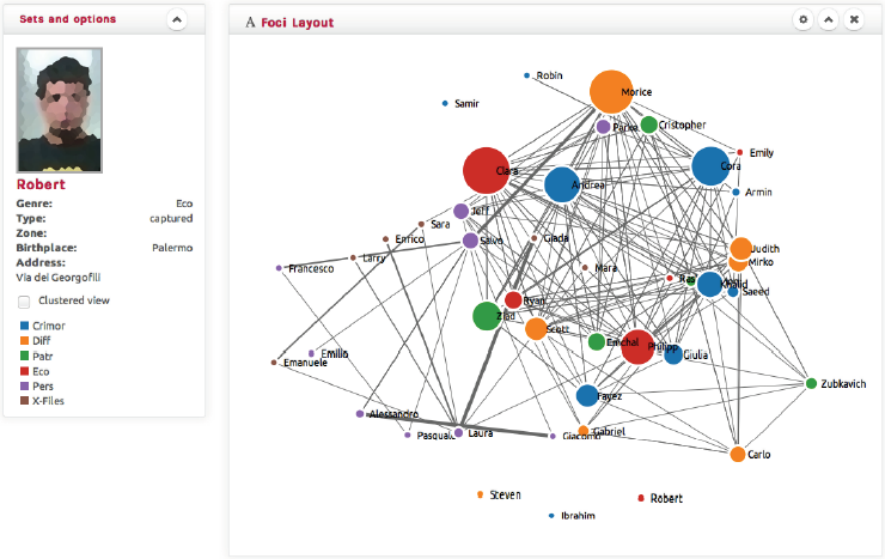


Fig. 1.4: Foci layout.

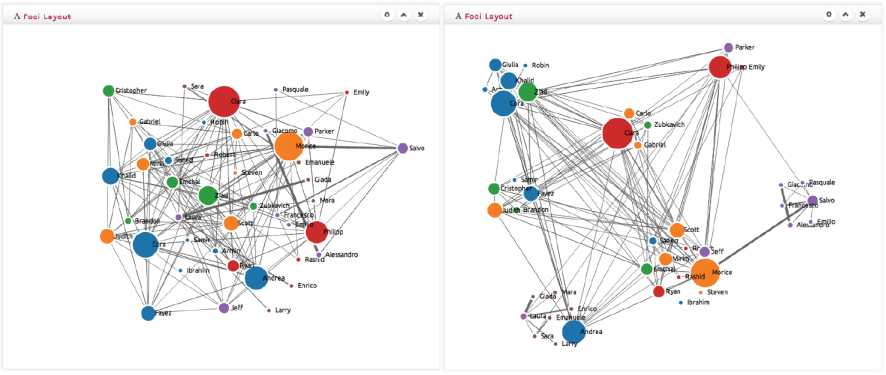


Fig. 1.5: Multi-foci layout.

1.4 Spatio-temporal criminal networks analysis

1.4.1 Temporal network analysis

Phone call records and online social network data comes with temporal information attached to many events. For example, the time and duration of a call or a chat

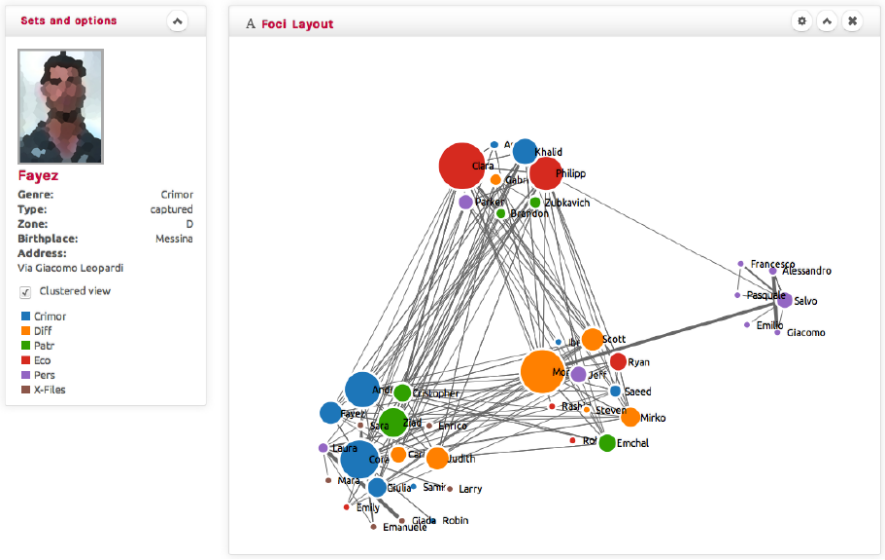


Fig. 1.6: Filtered and clustered multi-foci layout.

session, or the timestamp associated with the creation of a given phone contract or account on a social platform, are common meta-data available for investigation.

LogViewer provides extensive support to encode and exploit temporal information, when available, to perform network dynamic and temporal pattern analysis. One example is provided in Figure 1.7, where we display LogViewer's interface reporting aggregate temporal statistics related to the activity ongoing on a mobile phone network under investigation.

In this example three types of information are displayed: on top, a time series reports the volume of calls per day during the investigation period. It's possible to see how heterogeneous is this traffic, with a strong attenuation toward the end of the observation period, after a spike coinciding with an actual criminal event in the real world. The analysts has the possibility of zooming in the time series, to select different sub-intervals, to display different types of statistics over time (e.g., total volume of calls, or total duration, etc.) and to filter according to different types of constraints (e.g., showing only the information related to a subset of users, for example a particular cluster). The applied filters are also reported underneath, for example as pie charts that show specific statistics per day of the week, per type of event (e.g., phone calls, texts, video calls, etc.) and per geographic area. Better resolution is provided by histograms that bin the given statistics, say number of calls, per hour of the day.

Another example is provided in Figure 1.8 that shows a *stream graph* adopted to visualize a sequence of temporal events on an aggregate basis. Stream graphs show the potential of tools that provide dynamics and interactive data exploration. The

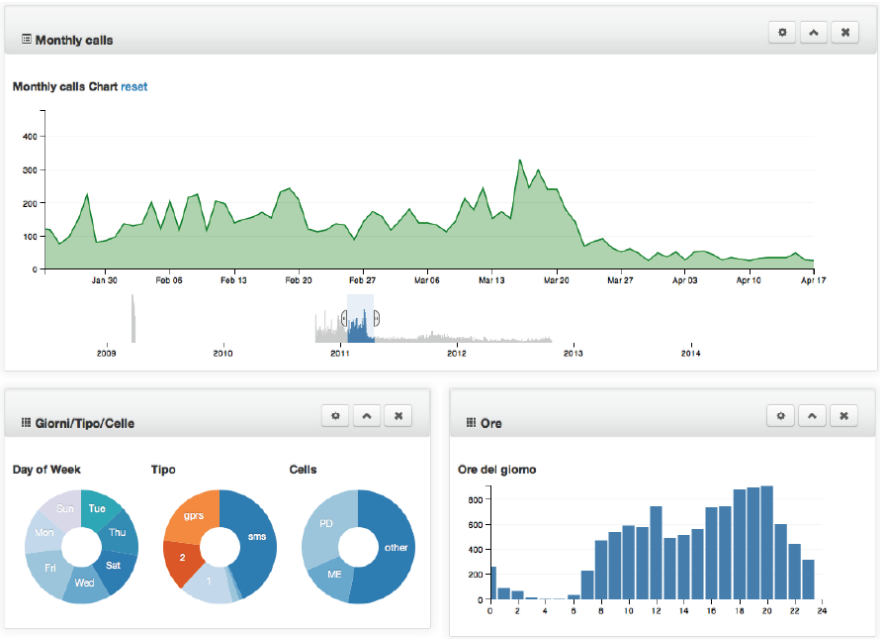


Fig. 1.7: Temporal analysis of a criminal network.

x axis of the stream graph represents time, whereas the *y* axis reports an arbitrary metric, say for the example in Figure 1.8 the total volume of phone interaction, subdivided by type (e.g., calls, texts, Internet sessions, etc.), each displayed using a different color. The stream is proportional to the number of events of each type per unit of time (one bin here is one hour). LogViewer also implements stacked graphs. Stream and stacked graphs represent especially helpful tools when the analysts want to visually compare extensive metrics that depend on the volume of events in a predetermined period.

By selecting the various temporal analysis tools and filters available, the analyst can dissect the dataset under analysis to obtain granular temporal information or to highlight and let emerge specific patterns of interactions among particular groups of individuals. This, in conjunction with spatial filters that are discussed in the next section, yields the ability to determine when (and where) information flows, and to identify the peeks and lows of interaction activity among the members of a criminal organization, to narrow down investigations towards specific periods of interest (that might concur with events in the real world).

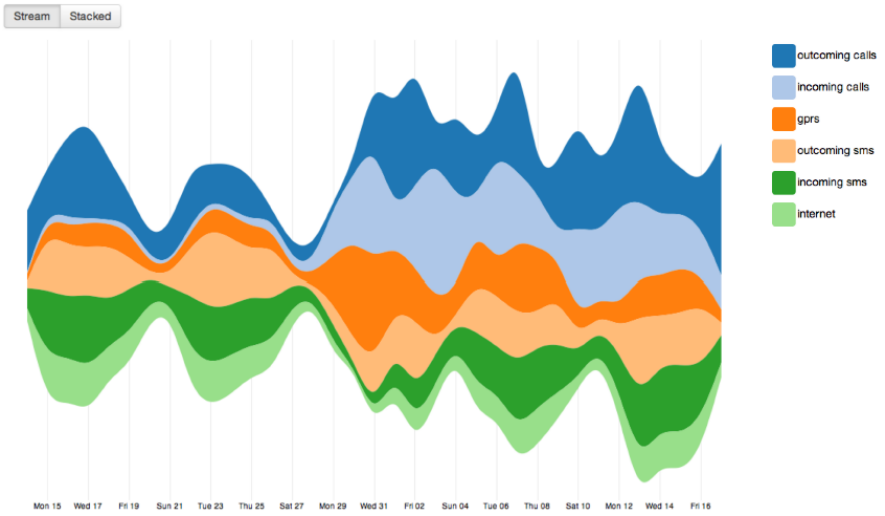


Fig. 1.8: Stream layout of temporal dynamics in a criminal network.

1.4.2 Spatial network analysis

Along with temporal information, phone call records and online network datasets report, among others, geographical coordinates of most of the events. Latitude and longitude can be inferred from the BTS (Base Transceiver Station) of a cell, or directly derived from the GPS sensors of enabled devices. In related work [23] we provide some additional detail on the inference mechanism behind the reconstruction of geo-coordinates from BTS cells.

LogViewer encodes, processes and presents spatial information to derive the mobility patterns of individuals, routine paths and points of interest, reconstructed from the geo-referenced interconnections (both phone calls and online social network sessions and check-ins). Figure 1.9 shows, for example, a case study inspired by a real investigation where nodes, displayed in overlay onto a map, represent areas where, during the observation period, intense contacts among a subset of the population under investigation took place (node sizes encode the volume of calls binned by geographic position). Different filters are provided, along with a slider that allows to "unfold" time and replay the evolution of such network simulating the temporal dimension. The spatial analysis, combined with the temporal filters, allow to observe the dynamic patterns of interconnections among the individuals under observation, and it's especially helpful to locate them in space and time, that could help in those cases when evidence is needed to prove someone's presence in a determined location during a specific event occurred in the real world (for example, a robbery or a homicide).

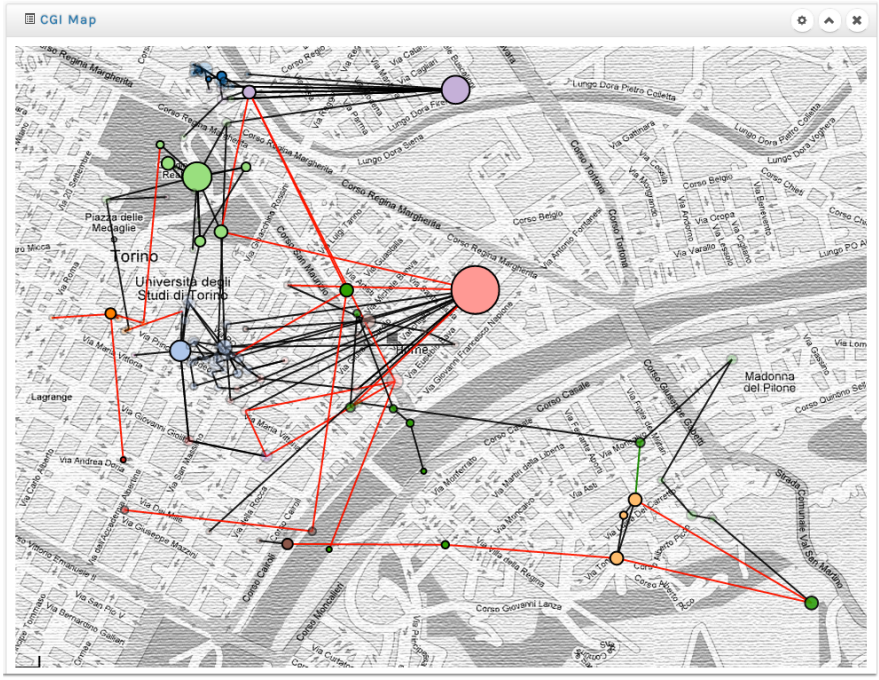


Fig. 1.9: Spatial analysis of a criminal network

1.5 A Use case inspired by real investigations

LogViewer has been successfully used in real forensic police investigations. Various examples, and the details of the analysis presented here, have been discussed in our latest work [23]: let us summarize few interesting results. Note that, as criminal lawsuits are still in progress, some information has been intentionally obfuscated.

1.5.1 The initial configuration

We here discuss a case in which some people allegedly belong to a criminal network. Police determined that phone traffic logs acquired (under court warrants) from the service providers of the suspects might reveal crucial information about their interpersonal relationships and communication dynamics. The logs reflect the phone calls occurred throughout fifteen days among these individuals allegedly part of a criminal association responsible of robbery, extortion and drug illicit trafficking.

From the analysis of the interactions occurred in a given time interval it is also possible to unveil the most important links, in terms of frequencies of relations and flow of information. Links do not necessarily reflect the same type of relations: different motivations can underlay phone interactions. In lack of advanced methods for conversation analysis (and due to the lack of phone call recordings), content analysis in some cases is impossible. However, the topology of the call networks is precious to reveal possible structural groups and, from there, ascertain the details.

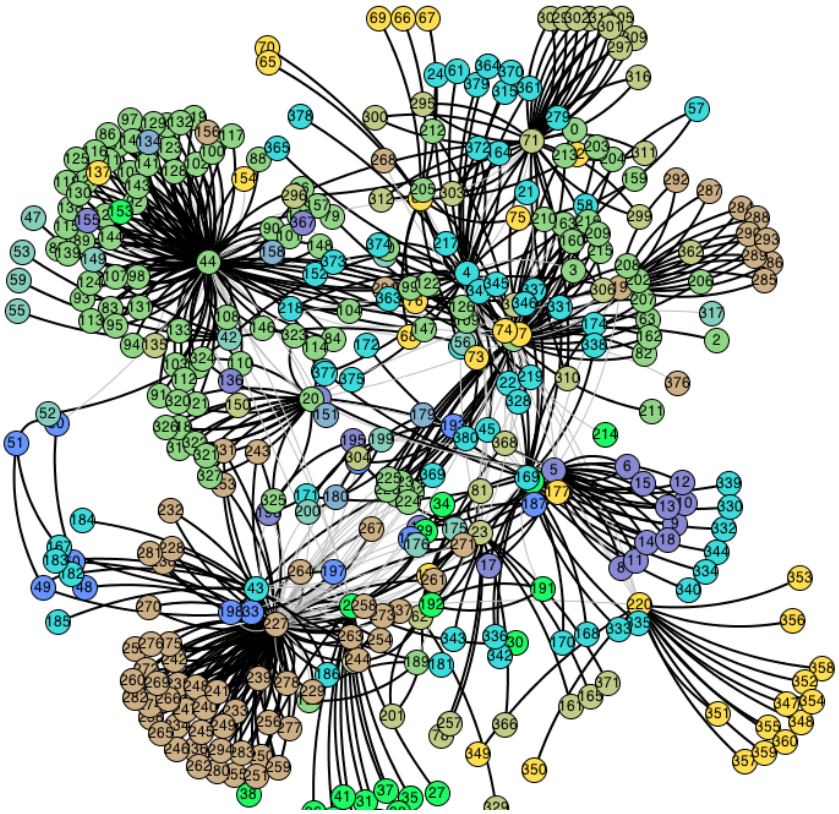
1.5.2 Finding subgroups

In Figure 1.10(a) we show the case study network after the Girvan-Newman (GN) algorithm has been executed and 16 communities have been detected. Different colors of the nodes identify different communities. To improve the clarity of the network visualization, we exploit the clustered view as shown in Figure 1.10(b). This configuration adopts a modified force-directed layout in which nodes of the same community (same colors) form macro-nodes visualized with a circular layout. In such a way, inter-connectivity among communities is captured better. The macro-nodes can be further exposed to reveal intra-community relationships (see Figure 1.10(c)).

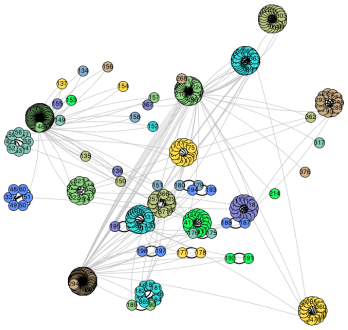
In this case, we are not interested only in the nodes that occupy prominent positions. Rather, we should focus on those edges whose deletion during the execution of the GN algorithm unveils new structural configurations, which in turn can be investigated using additional information available to police. This analysis will result of fundamental importance for the successful outcome of the investigation.

LogViewer supported this case investigation as follows: first, by automatically parsing interaction data (phone traffic) from heterogeneous sources; then, by abstracting a network representation of such data where nodes represent individuals, being links their interactions —a node-link layout is employed for visualization purpose. Finally, after performing community detection (and visualizing clusters), each member of these groups is analyzed in depth, recursively refining the results.

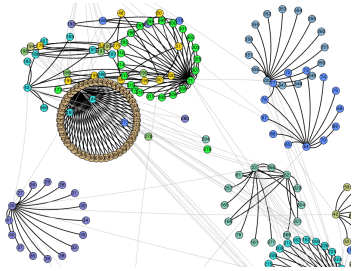
From clustering, two interesting results follow. First, the more central edges are not always responsible of driving the majority of the information, that is they are not in charge of communications among clusters. They are, however, still important edges from a topological point of view, and *lethal* when regarded inside their group. Secondly, clustering algorithms used to analyze criminal networks help to detect the tightest groups, but the nature of the relations must be carefully evaluated using information which can not be directly drawn from the mathematical model or its graphical representation. Network metrics applied to our case study reveal that the node with the highest degree (i.e., the highest number of phone calls) has a much lower betweenness centrality than other nodes. In fact, criminal networks heavily



(a) *Case study network after the GN algorithm. 16 communities have been detected.*



(b) *Clustered view. Nodes of the same community form macro-nodes visualized with a circular layout.*



(c) *Macro-nodes zoom reveals intra-community relationships.*

Fig. 1.10: Communities as obtained by using Girvan Newman algorithm.

employ secrecy to escape investigations and, in particular, a policy of internal communications according to which the most important members issue orders to a very limited number of members which in turn make them known to an increasing number of less important members until the leaves of the network are informed.

In our case study, the nodes having the highest number of communications (i.e., the highest degree) represent the lieutenants of the criminal organization and not the boss of the clan, while the edges traversed by the highest number of shortest paths (i.e., having the highest betweenness centrality) represent the most important links among the various groups.

Moreover, the granularity of the clustering allows to identify the optimal members and edges to remove when trying to hinder or disrupt the clan criminal activities.

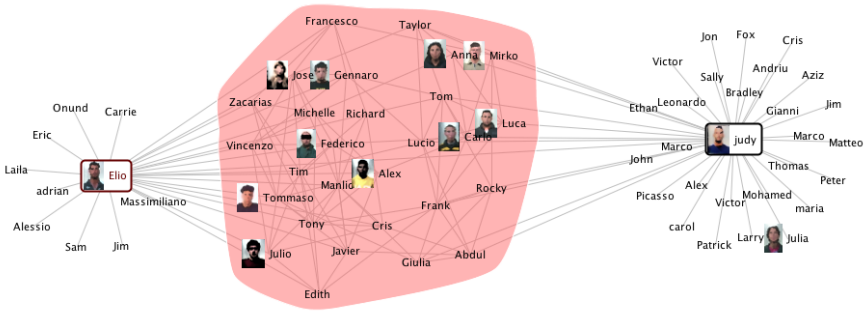
1.5.3 Overlapping communities

An important aspect in the analysis of communities is represented by the potential overlap of communities. Both the algorithms implemented in *LogViewer* actually perform a partition of the network, thus assigning each of the nodes to exactly one cluster. Often, this is not a correct representation, at least on a semantic basis, of the network. In a specific case such ours, even the algorithmic approaches described in [51, 60] may produce questionable results because of the multiplicity of meanings which can be given to any edge of the network. For this reasons, we decided to implement *LogViewer* in such a way to allow the user to choose the level of clustering in order to approximate the results. This feature is illustrated in Figure 1.11.

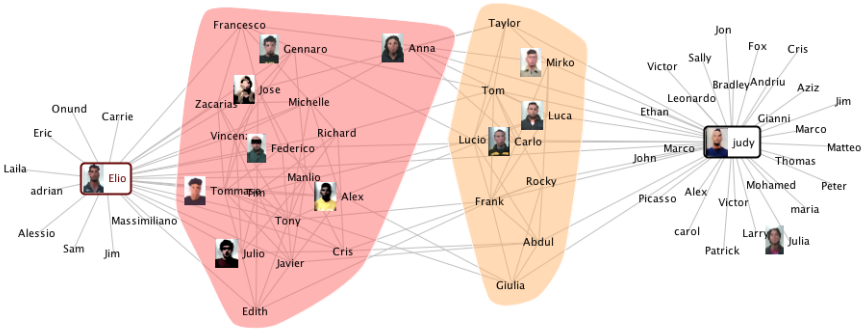
In Figure 1.11(a) only one cluster has been detected which is composed of the nodes interconnected among the external clusters represented by the nodes “Elio” and “Judy”, while in Figure 1.11(b), Q_{max} has been interactively decreased to a previous lower value. As a consequence, the interconnected nodes are subdivided and new communities emerge.

The in-depth analysis carried out on the members of the clusters interconnected (shown in Figure 1.11) —and the temporal analysis— allowed the investigators to discover that some clans belonging to the criminal network had worked with a certain degree of autonomy and were responsible of some murders. It turned out that these clans were tasked of committing murders on account of the organization. Figure 1.12 shows the clans at times t_1 and t_2 (all names are fictitious).

Some additional remarks are needed. Applying the GN community detection algorithm without supervision (i.e., only to maximize the modularity), produces a partition according to which the criminal network is composed of 14 clusters. The maximum partition density is 0.014 and the largest community is composed of 84 nodes. This clustering is not coherent with the real structural subdivision of the



(a) Q_{max} *modularity*.



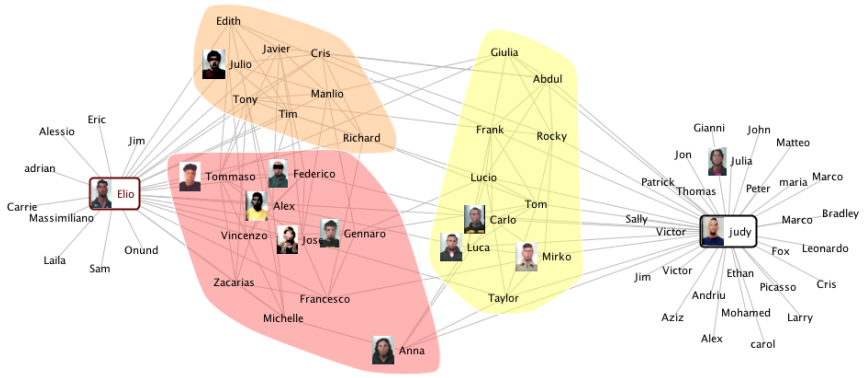
(b) $Q_{max} - 1$ *modularity*.

Fig. 1.11: An example of community detection using the Newman algortihm [48]. The convex-hull layout has been adopted for the visualization of the communities.

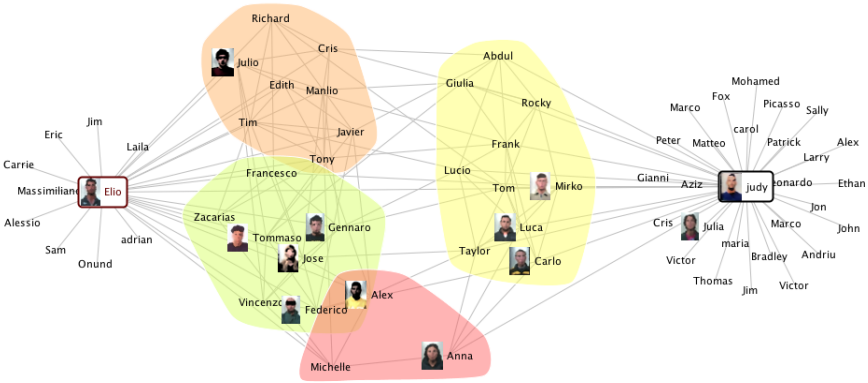
criminal network, as it emerged from the supervised interactive community detection procedure, combined with additional comparisons and in-depth examinations obtained from other informative sources. Nevertheless, this result was very interesting since important information regarding some members of the criminal network emerged.

In particular, from the analysis of the different levels of clustering selected interactively, and from the observation of the relative variations in the obtained configurations, we identified which elements of the network were affected mostly.

Concluding, the analysis of the distribution of phone calls carried out by each *clan* (see Figure 1.13) is a method generally very useful to decide if a good level of clustering has been obtained after the execution of the community detection algorithm. The goal of this analysis is twofold: first, it identifies the groups among which the largest number of phone calls, texts, MMS, etc., took place; second, it highlights the



(a) *The criminal network at time* t_1.



(b) *The criminal network at time* t_2.

Fig. 1.12: Community detection of a time-varying criminal network.

peaks of the stream of communications related not to single users but rather to each cluster as a whole, on the occasion of a crime.

1.6 Related Work

In the latest thirty years academic research related to the application of social network analysis to intelligence and study of criminal organizations has constantly grown. One of the most important studies is due to Malcolm Sparrow [59], related to the application of techniques of network analysis, and the study of network vulnerabilities, for intelligence scopes. He underlined three key aspects of so-called *crim-*

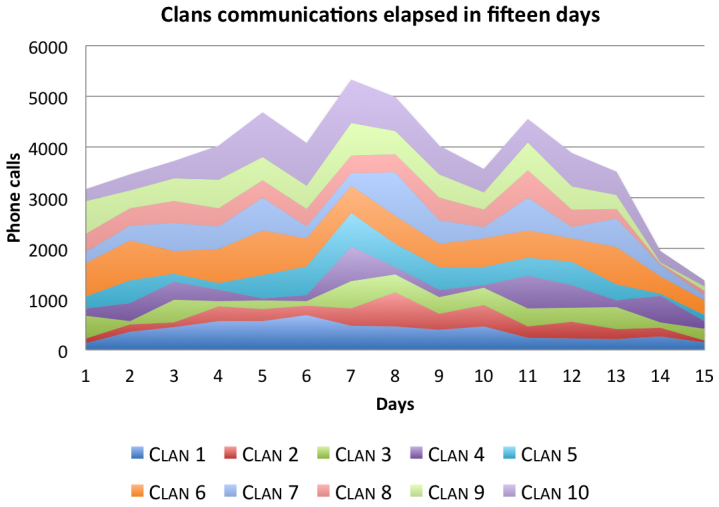


Fig. 1.13: Stacked histogram showing the phone call traffic carried out by each group (or clan) in the time interval of 15 days.

inal network analysis (CNA): i) the importance of *social network analysis* (SNA) for the analysis of criminal data; ii) the potential of added intelligence from network analysis and, iii) the results deriving from the collaboration between the two sectors.

Sparrow defined four features peculiar of criminal networks (CNs): i) limited dimension — CNs are often composed of at most few thousand nodes; ii) information incompleteness — criminal or terrorist networks are unavoidably incomplete due to fragmentary available information and erroneous information; iii) undefined borders — it is difficult to determine all the relations of a node; and, iv) dynamics — new connections imply a constant evolution of the structure of the network.

Thanks to Sparrow's work, other authors tried to study criminal networks using the tools of SNA. For example, Baker and Faulkner [3] studied illegal networks in the field of electric plants and Klerks [35] focused on criminal organizations in The Netherlands. In 2001, Silke [58] and Brennan et al. [10] acknowledged a slow growth in the fight against terrorism, and examined the state of the art in the field of criminal network analysis.

Arquilla and Ronfeldt [1] summarize prior research by introducing the concept of Netwar and its applicability to terrorism. They illustrate the difference between social networks and CNs, demonstrating the great utility of network models to understand the nature of criminal organizations. Their work shed light on strategies, methods and systems of information flow for intelligence purpose. The framework proposed by Arquilla and Ronfeldt provided new ground for conceiving network analysis. Nevertheless, they received some criticism due to their theoretical ap-

proach. Before 2001-09-11, some criticism can be found in the work of Carley, Reminga and Kamneva [11], devoted to destabilizing initiatives of dynamic terrorist networks.

All these early studies somehow neglected the importance of network visualization, stressing aspects related more to statistical network characterization, or interpretation of individuals' roles rooted in social theory. However, in 2006, a popular work by Valdis Krebs [36] applied graph analysis in conjunction with network visualization theory to analyze the Al Qaeda cell responsible of the 2001-09-11 terrorist attacks in the USA. This work represents a starting point of a series of academic papers in which social network analysis methods become applied to a real-world cases, differently from previous work where mostly toy models and fictitious networks were used. Krebs' paper is one of the more cited papers in the field of application of social network analysis to Criminal Networks and it inspired further research in network visualization for the design and development of better SNA tools applications to support intelligence agencies in the fight against terror, and law enforcement agencies in their quest fighting crime.

In criminology and research on terrorism, SNA has been proved a powerful tool to learn the structure of a criminal organization. It allows analysts to understand the structural relevance of single actors and the relations among members, when regarded as individuals or members of (one or more) subgroup(s). SNA defines the key concepts to characterize network structure and roles, such as centrality [25], node and edge betweenness [25, 8], and structural similarity [41]. The understanding of network structure derived from these concepts would not be possible otherwise [65]. The above-mentioned structural properties are heavily employed to visually represent social and criminal networks as a support decision-making processes.

SNA provides key techniques including the possibility to detect clusters, identify the most important actors and their roles and unveil interactions through various graphical representation methodologies [73]. Some of these methods are explicitly designed to identify groups within the network, while others have been developed to show social positions of group members. The most common graphical layouts have historically been the node-link and the matrix representations [26].

Visualization has become increasingly important to gain information about the structure and the dynamics of social networks: since the introduction of sociograms, it appeared clear that a deep understanding of a social network was not achievable only through some statistical network characterization [65]. For all these reasons, a number of different challenges in network visualization have been proposed [57]. The study of network visualization focuses on the solution of the problems related to clarity and scalability of the methods of automatic representation. The development of a visualization system exploits various technologies and faces some fundamental aspects such as: i) the choice of the layout; ii) the exploration dynamics; and, iii) the interactivity modes introduced to reduce the visual complexity.

Recent studies tried to improve the exploration of networks by adding views, user interface techniques and modes of interaction more advanced than the conventional node-link and force-directed [27] layouts. For example, in *SocialAction* [52] users are able to classify and filter the nodes of the network according to the values of their statistical properties. In *MatrixExplorer* [32] the node-link layout is integrated with the matrix layout. Nonetheless, these visualization systems have not been explicitly developed with the aim of the exhaustive comprehension of all properties of the network. Users need to synthesize the results coming from some views and assemble metrics with the overall structure of the network.

Therefore, we believe that an efficient method to enhance the comprehension and the study of social networks, and in particular of criminal networks, is to provide a more explicit and effective node-link layout algorithm. This way, important insights could be obtained from a unique layout rather than from the synthesis derived from some different layouts.

We recently presented a framework, called *LogAnalysis* [13, 23], that incorporates various features of social network analysis tools, but explicitly designed to handle criminal networks reconstructed from phone call interactions. This framework allows to visualize and analyze the phone traffic of a criminal network by integrating the node-link layout representation together with the navigation techniques of zooming and focusing and contextualizing. The reduction of the visual complexity is obtained by using hierarchical clustering algorithms. In this chapter we discuss three new network layout methods that have been recently introduced in *LogViewer*, namely fisheye, foci and geo-mapping, and we explain how these methods help investigators and law enforcement agents in their quest to fight crime.

It's worth noting that various tools to support network analysis exist. However, only few of them have been developed specifically for criminal network investigations. We mention, among others, commercial tools like COPLINK [14, 71], Analyst's Notebook[4], Xanalysis Link Explorer[5] and Palantir Government[6]. Other prototypes described in academic papers include Sandbox [68] and POLESTAR [53]. Some of these tools show similar features to *LogViewer*, but, to the best of our knowledge, none of them yields the same effective and scalable network visualization with support to criminal networks reconstructed from phone call records.

1.7 Conclusions

In this chapter we presented *LogViewer*, a next-generation Web-based framework that provides advanced features for criminal network analysis. We first provided a

[4] `ibm.com/software/products/analysts-notebook/`

[5] `http://www.xanalys.com/products/link-explorer/`

[6] `http://www.palantir.com/solutions/`

high-level overview of the workflow that analysts follow to bootstrap a criminal investigation by using a framework like ours, and then we presented some underlying theory behind the network measures, clustering methods, and visualization techniques adopted to uncover criminal behavior in spatio-temporal networks reconstructed from microscopic human interactions (e.g., mobile phone calls or online social network data).

LogViewer paves the way for the creation of a general framework for the identification of criminal activities from digital footprints, however there is a lot to be done yet. In our vision, this framework will extend at least in three fundamental directions in the future: (i) infer roles of individuals in the hierarchical structure of a criminal organization; (ii) predict crimes from spatio-temporal patterns of criminal activity; (iii) predict which individuals within a social network are more exposed to the possibility of turning into criminals in the future, given their social circles and their interactions with existing criminals.

Concluding, from a technical perspective, we are already working to incorporate further sources of network interactions at the microscopic level, such as financial transaction records or face-to-face interactions that might be recorded and tracked through advanced traditional investigation methods.

References

1. J. Arquilla and D. Ronfeldt. Networks and netwars: The future of terror, crime, and militancy. *Survival*, 44(2):175–176, 2001.
2. J. Assa, D. Cohen-Or, and T. Milo. Displaying data in multidimensional relevance space with 2d visualization maps. In *Proc. Visualization '97*, pages 127–134, 1997.
3. W. Baker and R. Faulkner. The social organization of conspiracy: illegal networks in the heavy electrical equipment industry. *Am. Social. Rev.*, 58:837–860, 1993.
4. E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic. The role of social networks in information diffusion. In *Proceedings of the 21st international conference on World Wide Web*, pages 519–528. ACM, 2012.
5. J. Barnes and P. Hut. A hierarchical 0(N log N) force-calculation algorithm. *Nature*, 324:4, 1986.
6. V. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, page P10008, 2008.
7. A. Bogomolov, B. Lepri, J. Staiano, N. Oliver, F. Pianesi, and A. Pentland. Once upon a crime: Towards crime prediction from demographics and mobile data. *arXiv preprint arXiv:1409.2983*, 2014.
8. U. Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25(2):163–177, 2001.
9. U. Brandes. Drawing on physical analogies. In M. Kaufmann and D. Wagner, editors, *Drawing Graphs*, volume 2025 of *Lecture Notes in Computer Science*, pages 71–86. Springer Berlin Heidelberg, 2001.
10. D. W. Brannan, P. F. Esler, and N. T. Anders Strindberg. Talking to terrorists: Towards an independent analytical framework for the study of violent substate activism. *Studies in Conflict and Terrorism*, 24(1):3–24, 2001.

11. R. J. Carley, K. M. and N. Kammneva. Destabilizing terrorist networks. *Institute for Software Research*, (45), 1998.
12. E. Casey. *Digital evidence and computer crime: forensic science, computers and the internet*. Academic press, 2011.
13. S. Catanese, E. Ferrara, and G. Fiumara. Forensic analysis of phone call networks. *Social Network Analysis and Mining*, pages 1–19, 2013.
14. H. Chen, D. Zeng, H. Atabakhsh, W. Wyzga, and J. Schroeder. Coplink: managing law enforcement data and knowledge. *Communications of the ACM*, 46(1):28–34, 2003.
15. F. Ciulla, D. Mocanu, A. Baronchelli, B. Gonçalves, N. Perra, and A. Vespignani. Beating the news using social media: the case study of american idol. *EPJ Data Science*, 1(1):8, 2012.
16. M. D. Conover, C. Davis, E. Ferrara, K. McKelvey, F. Menczer, and A. Flammini. The geospatial characteristics of a social movement communication network. *PloS one*, 8(3):e55957, 2013.
17. M. D. Conover, E. Ferrara, F. Menczer, and A. Flammini. The digital evolution of Occupy Wall Street. *PloS one*, 8(5):e64679, 2013.
18. M. D. Conover, B. Gonçalves, A. Flammini, and F. Menczer. Partisan asymmetries in online political activity. *EPJ Data Science*, 1:6, 2012.
19. P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti. Enhancing community detection using a network weighting strategy. *Information Sciences*, 222:648–668, 2013.
20. P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti. Mixing local and global information for community detection in large networks. *Journal of Computer and System Sciences*, 80(1):72–87, 2014.
21. P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti. On Facebook, most ties are weak. *Communications of the ACM*, 57(11):78–84, 2014.
22. E. Ferrara. A large-scale community structure analysis in Facebook. *EPJ Data Science*, 1(9):1–30, 2012.
23. E. Ferrara, P. De Meo, S. Catanese, and G. Fiumara. Detecting criminal organizations in mobile phone networks. *Expert Systems with Applications*, 41(13):5733–5750, 2014.
24. E. Ferrara, O. Varol, F. Menczer, and A. Flammini. Traveling trends: social butterflies or frequent fliers? In *Proceedings of the first ACM conference on Online social networks*, pages 213–222. ACM, 2013.
25. L. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.
26. L. C. Freeman. Visualizing social networks. *Journal of Social Structure*, 1, 2000.
27. T. Fruchterman and E. Reingold. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.
28. G. W. Furnas. Generalized fisheye views. *SIGCHI Bull.*, 17(4):16–23, Apr. 1986.
29. M. Girvan and M. Newman. Community structure in social and biological networks. *Proc. Natl. Acad. Sci.*, 99(12):7821, 2002.
30. B. Gonçalves, N. Perra, and A. Vespignani. Modeling users' activity on Twitter networks: Validation of dunbar's number. *PloS one*, 6(8):e22656, 2011.
31. S. González-Bailón, J. Borge-Holthoefer, A. Rivero, and Y. Moreno. The dynamics of protest recruitment through an online network. *Scientific reports*, 1, 2011.
32. N. Henry and J.-D. Fekete. Matrixexplorer: A dual-representation system to explore social networks. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):677–684, Sept. 2006.
33. M. Hypponen. Malware goes mobile. *Scientific American*, 295(5):70–77, 2006.
34. Y. Jewkes and M. Yar. *Handbook of Internet crime*. Routledge, 2013.
35. P. Klerks and E. Smeets. The network paradigm applied to criminal organizations: Theoretical nitpicking or a relevant doctrine for investigators? recent developments in the netherlands. *Connections*, 24:53–65, 2001.
36. V. Krebs. Mapping networks of terrorist cells. *Connections*, 24(3):43–52, 2002.
37. R. Kumar, J. Novak, and A. Tomkins. Structure and evolution of online social networks. In *Link mining: models, algorithms, and applications*, pages 337–357. Springer, 2010.

38. N. Leavitt. Mobile phones: the next frontier for hackers? *Computer*, 38(4):20–23, 2005.
39. J. Lehmann, B. Gonçalves, J. J. Ramasco, and C. Cattuto. Dynamical classes of collective attention in Twitter. In *Proceedings of the 21st international conference on World Wide Web*, pages 251–260. ACM, 2012.
40. Y. K. Leung and M. D. Apperley. A review and taxonomy of distortion-oriented presentation techniques. *ACM Trans. Comput.-Hum. Interact.*, 1(2):126–160, June 1994.
41. F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *The Journal of Mathematical Sociology*, 1(1):49–80, 1971.
42. A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 29–42. ACM, 2007.
43. D. Mocanu, A. Baronchelli, N. Perra, B. Gonçalves, Q. Zhang, and A. Vespignani. The Twitter of babel: Mapping world languages through microblogging platforms. *PloS one*, 8(4):e61981, 2013.
44. C. Morselli. *Contacts, opportunities, and criminal enterprise*. University of Toronto Press, 2005.
45. C. Morselli. *Inside criminal networks*, volume 8. Springer, 2008.
46. C. Morselli. Assessing vulnerable and strategic positions in a criminal network. *Journal of Contemporary Criminal Justice*, 26(4):382–392, 2010.
47. S. A. Myers, C. Zhu, and J. Leskovec. Information diffusion and external influence in networks. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 33–41. ACM, 2012.
48. M. Newman. Fast algorithm for detecting community structure in networks. *Phys. Rev. E*, 69(6):066133, 2004.
49. M. Newman. A measure of betweenness centrality based on random walks. *Social Networks*, 27(1):39–54, 2005.
50. M. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phy. Rev. E*, 69(2):26113, 2004.
51. Palla, G. and Derényi, I. and Farkas, I. and Vicsek, T. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435:814–818, 2005
52. A. Perer and B. Shneiderman. Balancing systematic and flexible exploration of social networks. *IEEE Transactions on Visualization and Computer Graphics*, pages 693–700, 2006.
53. N. J. Pioch and J. O. Everett. Polestar: collaborative knowledge management and sensemaking tools for intelligence analysts. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 513–521. ACM, 2006.
54. D. M. Romero, B. Meeder, and J. Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on Twitter. In *Proceedings of the 20th international conference on World wide web*, pages 695–704. ACM, 2011.
55. M. Sageman. *Understanding Terror Networks*. University of Pennsylvania Press, 2004.
56. M. Sarkar and M. H. Brown. Graphical fisheye views. *Comm. ACM*, 37(12):73–84, 1994.
57. F. Schneider, A. Feldmann, B. Krishnamurthy, and W. Willinger. Understanding online social network usage from a network perspective. In *Proceedings of the 9th SIGCOMM conference on Internet measurement conference*, pages 35–48. ACM, 2009.
58. A. Slike. The devil you know: Continuing problems with research on terrorism. *Terrorism and Political Violence*, 13:1–14, 2001.
59. M. K. Sparrow. The application of network analysis to criminal intelligence: An assessment of the prospects. *Social Networks*, 13(3):251–274, 1991.
60. Sun, P.G. and Gao, L. and Shan Han, S. Identification of overlapping and non-overlapping community structure by fuzzy clustering in complex networks. *Information Sciences*, 181:1060–1071, 2011
61. M. Todd and A. Nomani. *The Truth Left Behind: Inside the Kidnapping and Murder of Daniel Pearl*. New York (2011) - http://www.publicintegrity.org/2011/01/20/2190/, 2011.
62. O. Varol, E. Ferrara, C. Ogan, F. Menczer, and A. Flammini. Evolution of online user behavior during a social upheaval. In *Proceedings of the 2014 ACM conference on Web Science*, pages 81–90. ACM, 2014.

63. A. Vespignani. Predicting the behavior of techno-social systems. *Science*, 325(5939):425, 2009.
64. X. Wang, M. S. Gerber, and D. E. Brown. Automatic crime prediction using events extracted from Twitter posts. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, pages 231–238. Springer, 2012.
65. S. Wasserman and K. Faust. *Social network analysis: methods and applications*. Cambridge University Press, 1994.
66. L. Weng, A. Flammini, A. Vespignani, and F. Menczer. Competition among memes in a world with limited attention. *Scientific Reports*, 2, 2012.
67. U. K. Wiil, J. Gniadek, and N. Memon. Measuring link importance in terrorist networks. In N. Memon and R. Alhajj, editors, *ASONAM*, pages 225–232. IEEE Computer Society, 2010.
68. W. Wright, D. Schroh, P. Proulx, A. Skaburskis, and B. Cort. The sandbox for analysis: Concepts and methods. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 801–810, New York, NY, USA, 2006. ACM.
69. J. Xu and H. Chen. Criminal network analysis and visualization. *Communications of the ACM*, 48(6):100–107, 2005.
70. J. Xu, B. Marshall, S. Kaza, and H. Chen. Analyzing and visualizing criminal network dynamics: A case study. In *Intelligence and Security Informatics*, pages 359–377. Springer, 2004.
71. J. J. Xu and H. Chen. Crimenet explorer: a framework for criminal network knowledge discovery. *ACM Transactions on Information Systems (TOIS)*, 23(2):201–226, 2005.
72. C. Yang, H. Chen, and K. Hong. Visualization of large category map for internet browsing. *Decis. Support Syst.*, 35(1):89–102, Apr. 2003.
73. C. C. Yang, N. Liu, and M. Sageman. Analyzing the terrorist social networks with visualization tools. In *ISI*, volume 3975 of *Lecture Notes in Computer Science*, pages 331–342. Springer, 2006.