Robinson and Scogings *Secur Inform (2018) 7:2*
https://doi.org/10.1186/s13388-018-0031-9

Security Informatics

RESEARCH

Open Access

The detection of criminal groups in real-world fused data: using the graph-mining algorithm "GraphExtract"

David Robinson* and Chris Scogings

Abstract

Law enforcement and intelligence agencies generally have access to a number of rich data sources, both structured and unstructured, and with the advent of high performing entity resolution it is now possible to fuse multiple heterogeneous datasets into an explicit generic data representation. But once this is achieved how should agencies go about attempting to exploit this data by proactively identifying criminal events and the actors responsible? The authors will outline an effective generic method that; computationally extracts minimally overlapping contextual subgraphs, then uses these subgraphs as the basis to construct a mesoscopic graph based on the intersections between the subgraphs, enabling knowledge discovery from these data representations for the purpose of maximally disrupting terrorism, organised crime and the broader criminal network.

Keywords: Graph mining, Criminal analysis, Crime, Organised crime, Criminal networks, Dark networks, Entity resolution, Risk detection, Social network analysis, SNA

Background

Traditionally law enforcement and intelligence agencies have relied on reactive sources of crime detection, such as the receipt of suspicious transactions, 'tip offs' from a covert human intelligence source (CHIS), or some other significant event (e.g. border interaction, search warrant, etc.). These reactive sources of detecting crime rely on human intervention. But now law enforcement and intelligence agencies commonly have access to a range of relevant large disparate datasets, which when fused can be used to proactively detect fragments of crime [1]. For example, a financial crime intelligence capability would be expected to have access to criminal data, corporate registers, asset registers, and reports of suspicious transactions. This results in fused data in the millions of records. Rather than using the data as a reference asset for simple querying law enforcement and intelligence agencies can now apply more proactive methods to infer and discover new knowledge from the data. This creates the opportunity to not only detect previously unobserved instances of crime, but also generate a contextual systems view of those crime instances.

In the specific area of crime detection law enforcement and intelligence agencies generally deploy anomaly detection, supervised learning or unsupervised learning techniques that focus on narrow sub-types of crime that are well historically understood, have a large set of examples, and the quality of data is well appreciated. These approaches fundamentally rely on a closed world approach constraining the problem to a very specific concise conceptual level (e.g. prediction of a sub-populations involvement in a very specific financial crime) or rely on an entity to seed the problem (e.g. find all shortest paths between the source entity and a target entity) [2, 3]. An alternative approach is using the entire context of the criminal complex system to support the detection of crime fragments, adopting the open world assumption. The assumption that the data is only ever a partial 'dirty' representation of the real-world is important for law enforcement and intelligence agencies. The domain includes significant misinformation such as fake entities, name variants, and multiple phone usage [4], and a

*Correspondence: david.robinson.18@uni.massey.ac.nz
Institute of Natural and Mathematical Sciences, Massey University, Auckland, New Zealand

primary effort to obfuscate identity, in addition to significant data incompleteness.

The term complex systems is used specifically as criminal actors naturally coalesce together into functional groups, exhibit different sets of capital and knowledge and influence one another [5–7], conduct criminal acts in concert with each actor fulfilling specific roles [8–11], within the broader context of a complex system [1].

This paper outlines a novel graph mining method—the "GraphExtract" algorithm—that detects overlapping subgraphs (i.e. a collection of nodes and edges) of entities involved in atomic criminal events and generates a contextual view of how these subgraphs are connected to one another.

At this point it is important to state that the detection of criminal subgraphs is on a prima facie basis as given the data and context we cannot possibly know for sure, at this early stage, whether the detected subgraph either represents anywhere near the complete subgraph, nor whether the subgraph represents an actual criminal event. This can only be known when sufficient evidence is collected and collated. The word atomic is used as it is clear that criminal acts, and particularly profit-driven criminal acts, are interdependent (e.g. the illicit drug supply chain) and form a complex system [1]. So, any attempt to identify a criminal event needs to be undertaken within this broad context.

The value of utilising such a computational approach to ally traditional methods of crime detection is clear. Firstly, any well designed and implemented computational approach should bring with it the benefits of being repeatable, consistent, measurable, extensible, scalable, efficient, transparent, and with the opportunity for continuous improvement, in comparison to any manual or solely reactive approach. Secondly, coverage of the entire criminal spectrum becomes possible and not limited to the reach of particular reactive pathways each agency has at their disposal. For example, within English speaking countries the border domain CHIS approaches may be only reasonably successful with English speaking groups, but largely neglect groups that speak a foreign language. Thirdly, "GraphExtract" does not rely on training data and therefore is more widely applicable than supervised learning approaches. Fourthly, detecting a range of subgraphs creates the opportunity to understand how these subgraphs are connected within the broader context of the entire system. This enables better decision-making on what criminal instances to focus on and how to apply resource (e.g. surveillance, phone record requisition).

Next we will outline the core technology required to address the data challenges, cover related work and then within this literature context describe the "GraphExtract" algorithm.

Core technology required to address the data challenges

The data needs to be fundamentally represented in a flexible way that enables framing the problem as a complex system. The solution to this is adopting a graph [12] representation that enables a more expressive data model that can better capture the embedded contextual relationship features that is the signature of criminal activity. As mentioned earlier the reality is that the incomplete 'poor' quality data represented in a disparate collection of datasets has to be addressed to reduce uncertainty to the point that graph mining generates value.

Entity resolution is the critical technology to fuse datasets that do not have unique identifiers that otherwise can be easily "joined". Entity resolution focuses on the identification of instances of where real-world entities (e.g. a person or company) are represented multiple times across the collection of datasets [13], and high performance entity resolution is critical within this domain [14].

Link Discovery is a subset of link and node discovery, which is further defined as the inference and prediction of unknown edges and nodes. Within the criminal domain this step is fundamental as graphs will not just be incomplete with missing nodes and edges, but the data will also include fake and spoof nodes. Fake nodes are nodes in the dataset but not in the real world, and spoof nodes are real world nodes that are represented as non-identical nodes in the data [4]. Link Discovery in particular is a critical element to enhancing the quality and completeness of criminal data [15, 16].

Related work

The challenges are clear—there is a partial 'dirty' high uncertainty large disparate collection of datasets available to provide a window into the real world of crime. Utilisation of technology such as graph representations, entity resolution, and link prediction creates the opportunity to then use this data to proactively detect fragments of criminal activity. This goal is far from the discrete efforts to use supervised or unsupervised learning to identify specific well understood narrow instances of crime, which has a basis of data couched in a closed world. The "GraphExtract" algorithm is designed to operate very much at the start of the intelligence and investigative processes. This is when data is most sparse, variable, and low on detail, aiming to take a proactive approach to detecting atomic instances of crime.

There is no analogous method within the literature. So on what basis do we define related work of an algorithm that has such a generic goal, where only a fraction of approaches are detailed in the literature and the residual are proprietary and therefore not open to academic scrutiny? The scope of related work has been created by articulating the "GraphExtract" algorithm firmly as a graph mining approach. Having done this we will firstly cover generic graph mining approaches most analogous to the "GraphExtract" algorithm. Then subsequent to this we will cover a number of published graph based computational solutions that focus on the criminal domain. The goal here is applying a repeatable computational approach on a relatively large set of fused non-curated datasets (e.g. 10 m + node graph), which is quite a different proposition than testing an approach on a relatively small set of well curated data (e.g. 50–50,000 node graph). The two pathways, addressing graph mining and computational graph solutions in the criminal domain, in combination provide a strong contextual platform to contrast and understand how the "GraphExtract" algorithm is novel, both in design and application, in terms of published literature. However, this must be couched in the context of the reality that law enforcement and intelligence agencies traditionally do not publish methodology openly.

Graph mining can be defined as simply the detection of patterns in graphs. Here we are narrowing the term somewhat to describe the identification of subgraphs of interest from a graph. So, that each subgraph has to be relevant. Furthermore, in our case each subgraph is also likely to naturally have an overlap with adjacent subgraphs. A number of generic graph methods can be applied to achieve this, including frequent subgraph mining and clustering, so let's introduce these in turn.

Frequent subgraph mining (FSM) uses subgraph structure as the predominant feature from which pattern detection is based, either in the case of identifying subgraph isomorphisms [17] or inferring specific topological patterns [18].

Clustering is the task of placing nodes in the graph into specific classes ensuring that nodes that are similar are given the same class. Importantly, the relevant sub-type of clustering task here should allow singleton class nodes (i.e. no class attributed to a node) and allow overlapping classes [19]. Alternatively, structural and regular equivalence form a specific set of clustering methods, usually deployed using blockmodelling or stochastic blockmodelling, which can conceptually detect graph structure and specifically classes of role [20].

FSM and clustering, although not considered supervised, suffer from the uncertainty and incompleteness present in the data, not to mention the scale of the data. FSM, clustering and specifically blockmodelling relies heavily on well curated data with explicit semantic edge labels within a constrained data context. Xu and Chen [21] successfully deployed blockmodelling, and a range of social network analysis (SNA) metrics, to criminal graphs of sizes 60 and 57 nodes. They demonstrated both the potential value of such approaches on small subgraphs and the computational limitations of such approaches with large graphs.

What graph mining computational approaches have been applied to the criminal domain, and how are they fundamentally different from "GraphExtract"?

The COPLINK software originally developed from research between Arizona State University, Tuscon Police Department, and Phoenix Police Department since 1997, has contributed a significant body of literature on using SNA and related technology to increase our understanding of crime [22]. COPLINK does not focus on the detection of criminal subgraphs per se, but conducts a range of topological metrics on subgraphs after they have been identified [21, 23] including link analysis [3, 22], topology [15], and identification of significant facilitators in evolving criminal networks [16]. The examples provided by Xu and Chen [21] include two graphs of 57 and 60 nodes.

Another software product designed specifically for law enforcement and intelligence agencies is GANG [24]. Interestingly, GANG takes a known criminal group and partitions the group using the Louvain community detection algorithm [25] to then also give an 'ecosystem' view of how these sub-groups interconnect. Shakarian et al. [24] evaluated GANG on a 1468 node graph. So, GANG takes a small criminal subgraph as an input and then provides SNA metrics, including community detection, on that criminal subgraph. This is quite distinct to "GraphExtract". GANG does not detect criminal groups or subgraphs, it merely provides metrics on predefined criminal groups.

Graph mining approaches have been used to detect suspicious sets of transactions through graph isomorphism [26], detecting fraudulent behaviour using graph based anomaly detection (GBAD) [27, 28], and role based approaches have been used to detect terrorist groups [29]. These approaches are not applicable in the stated context due to the uncertainty and incompleteness present in the data, not to mention the scale of the data, which is significant enough to preclude the use of such supervised learning approaches. In any case, these are specific approaches to detect specific criminal graph patterns, whereas the goal here was to develop a conceptually broader generic approach that identifies subgraphs that represent fragments of generic criminal activity.

Ozgul et al. [30] developed a 'combined detection model for criminal network detection' (ComDM)—a combination of previous models (GDM, OGDM, and SoDM) developed by the same researchers. This set of models are supervised learning based and ComDM depends on rich criminal offence data, including co-offenders, crime location, temporal data, modus operandi, geographical data, and offender name similarity. Wang et al. [31] outline an interesting space clustering method to identify crime series committed by the same individual or group. However neither of these approaches are analogous to "GraphExtract" as both approaches are supervised learning approaches relying on 'having a database of crimes' and focusing on clustering known criminal events.

In 2017 Li et al. [32] developed an interesting approach to detect groups of entities involved in money laundering (ML). They used a temporal-directed version of the Louvain algorithm [25], implemented and optimised on Apache Spark using GraphX. The unsupervised approach is premised on building a multi-edge (i.e. dyads can share multiple edges) graph from transactional data, simplifying this multi-edge graph into a weighted simple graph (i.e. dyads can share only a single edge). Then filtering dyads out that only have a single transaction edge between source node and destination node. Maximal connected subgraphs are then identified, with each subgraph further partitioned using a temporal-directed version of the Louvain algorithm, specifically tuned to the domain of ML. Each community derived through this approach is ranked based on a set of weighted rules that include community aggregations of: number of nodes, number of edges, sum of money, average node degree, and temporal entropy. Core assumptions noted include communities that are less complex are more likely ML gangs and that hub nodes indicate a higher possibility of ML. This solution may be scalable and performant on the specific case outlined however it is tightly coupled to the domain. The approach is not widely applicable outside the ML domain. Furthermore, community detection algorithms performance degrades with dense graphs [33]. This performance degradation exposes any subgraph detection solution that is dependent on community detection to extreme performance variability across graphs of differing topology.

The remainder of the criminal focussed graph based literature focuses on hypothesising how specific SNA metrics can be useful in generating knowledge about a small criminal subgraph via case study. In these instances the criminal network under consideration is post-detection—i.e. it has already been detected and curated for knowledge discovery. For example, Carley et al. [34] work on destabilising terror networks notes scalability of their approach is limited to graphs of 1000's of nodes, Krebs [35] focussed his SNA analysis on the 9/11 network (37 nodes), in 2010 Morselli [36] conducted SNA on a range of criminal groups ranging in size from 25 nodes through to 174 nodes, Everton [37] provided a case study on the Noordin Top terrorist group (79 nodes), and Morselli et al. [38] studied network stability of a network generated through the co-offending of 113,000 nodes. The baseline assumption of this body of research is that there is small well defined discrete well curated criminal network to apply a range of SNA metrics too.

So, now having covered law enforcement and intelligence agencies context, the data challenges, the potential value, and surveyed related approaches the challenge, purpose and novel value is clear. "GraphExtract" needs to:

- identify relevant fragments of overlapping criminal subgraphs,
- at the most atomic level,
- from large fused data (i.e. applicable on graphs over 10 million nodes) that represent non-criminal and criminal entities,
- given the data will be a 'dirty' partial representation,
- using a generic widely applicable method.

"GraphExtract" outline

So, with this context we will now outline the "GraphExtract" algorithm—a novel graph-mining solution. The "GraphExtract" algorithm takes a multi-modal graph (a graph with multiple node types—e.g. Person, Bank Account, Organisation, Phone, Address), fused from multiple datasets that includes criminal and non-criminal entities, as an input—let's call this input graph the original fused graph. The algorithm takes this original fused graph and labels each node based on that nodes role in profit-driven criminal activity. For example, a node is labelled "Predicate offence" if that node has been involved or alleged to be involved in the generation of illicit proceeds (e.g. drug importation). Other labels types include primary—"Associated offence", "Alleged money laundering offence", "Potential non-transparent money laundering vehicle"—and secondary—"Potential money laundering vehicle" and "Realised asset". Edges are labelled in terms of whether they satisfy the definition of trust or non-trust (see below for detail). The "entities of interest" set of nodes, made up of primary labelled nodes and relevant secondary labelled nodes, are partitioned into non-overlapping subsets (groups) based on their pairwise graph distance and subsequent community detection (see below for detail).

Mediating nodes for each of these subsets of "entities of interest" nodes are identified and included. Each subset of nodes is then used as the seeds from which to extract induced subgraphs from the original fused graph. The method to determine which nodes are included within each extracted subgraph is an iterative based neighbourhood approach that iteratively subsumes neighbours (excluding supernodes) terminating after either four hops have been completed or the size of the graph exceeds

150 entities. The subgraph is then pruned. The set of subgraphs is then represented as a mesoscopic graph with each subgraph represented as a node and the edges derived from the amount of intersection between each pair of subgraphs. The output of the "GraphExtract" algorithm includes a set of criminal subgraphs, referred to as the microscopic view, and a mesoscopic graph outlining how each criminal subgraph is connected across the entire criminal network.

The microscopic subgraphs and mesoscopic graph can then be targeted for knowledge discovery, creating further contextual knowledge, before being visualised and presented to users (e.g. intelligence analysts, investigators, managers, etc.).

Now that we have established the context of what the "GraphExtract" algorithm is and the value that such an approach can generate, we will build on this in the following sections. We will do this by detailing the core underpinning assumptions enabling the "GraphExtract" algorithm, before detailing the detailed design of the algorithm. This will be followed by an evaluative case study outlining how this approach works in a real-world setting assessing the methods performance, and rounded off by a conclusion identifying future extensions.

Assumptions and design of the "GraphExtract" algorithm

The "GraphExtract" algorithm outlined above has been developed to work effectively in the applied settings that law enforcement and intelligence agencies encounter. These applied settings absolutely require scalability and the acknowledgement of the data incompleteness and open world assumption. Thus, performance of the "GraphExtract" approach is dependent on a number of underpinning assumptions having been met.

Applied assumptions

Firstly, the collection of datasets must represent the core conceptual aspects of the problem. So, for example in the profit-driven crime domain datasets may represent the concepts of corporate ownership, non-transparency, assets/liabilities, and risk, across two dimensions—edges and attributes. Edges refer to where there is some sort of relationship between a pair of entities (e.g. a person has a shareholding in a company). Attributes refers to where an entity or relationship has some characteristic of interest (e.g. entity attribute—a person entity has the date of birth 01-01-1996; relationship attribute—the person has 100 shares in the company). So, the source datasets will hold relevant relational data about a range of relevant node types such as persons, companies, property, phones, bank accounts, etc. Having a range of datasets that satisfy the core conceptual aspects of the criminal act establishes the ability to measure the degree to which differing criminal elements are represented within each subgraph. For example, a subgraph that includes organised crime entities, a domestic corporate structure, a non-transparent offshore corporate structure, a series of assets owned by related parties, and a series of suspicious transaction reports can give an indication that a constellation of elements are present that represent the illicit generation, laundering, and realisation of proceeds. This is based on the premise that organised crime entities provide the access/means to generate illicit proceeds, corporate structures (particularly those non-transparent offshore structures) provide the means to launder proceeds, assets owned by related parties may have been realised illicitly, and suspicious transaction reports are indicators of money laundering.

As the data is a mere fragment of the real-world activity we can only hope to get partial views of this generation/launder/realisation process. The goal is to detect a kernel of criminal event sub-elements at the earliest instance enabling a contextual decision on where to focus resource (e.g. data collection, surveillance, etc.) and thereby reduce uncertainty allowing an informed decision on how to mitigate that criminal activity.

The second assumption is that the output of the entity resolution that fuses the disparate datasets together is represented as predicted relationships, with associated meta-data (e.g. a prediction [0–1]) on the quality of the prediction. This enables enhanced in situ decision-making on entity resolution. It is important to note that this class of edge is special in two key ways. Firstly, these predictions can be represented as contracted nodes or linked nodes, dependent on the context (they are represented as edges here). Secondly, entity resolution prediction is the cornerstone of accurate detection of these subgraph fragments and so retaining explicit visibility of the uncertainty coupled to the prediction is fundamental from both a modelling and a consumer perspective. This is not to imply other edge types do not have quantified uncertainty, however entity resolution predictions are of paramount significance.

The third assumption is that there is a graph data model and data quality that enables the identification of trust and non-trust relationships with accuracy. Non-trust relationships, defined here, are characterised by asymmetric relationships that are formed for a non-enduring transactional purpose, where there is no associated transfer of social capital engendering reciprocity. For example, a person undertaking a transaction at the casino does not in itself infer a meaningful relationship between that person and casino. Non-trust edges do not denote any substantial enduring or meaningful relationship, other than for the purpose of the single transaction, and so these relationships should be represented appropriately within the data model. Clearly not all relationships are the same and so failure to discern between trust and non-trust relationships will reduce the accuracy of defining the boundaries of subgraphs, as entities will be connected where in reality the relationship is trivial. So, it is critical to accurately identify and treat non-trust relationships appropriately, to reflect a more accurate representation of the real-world.

The fourth assumption is that the consumers of the output of this approach have a wide variety of goals based in the context of their role. Therefore, the "GraphExtract" algorithm attempts to maintain the open world stance to the latest possible moment. To create a generic method that satisfies the diverse range of users requirements means that the method has to focus on the most atomic set of entities required to operate together to represent an instance of generating, laundering and realising criminal proceeds. The key is to making decisions as late as possible with as much context as possible by those users with the right expertise and role.

Let's now get into a detailed view of how the "GraphExtract" algorithm is designed.

"GraphExtract": identify entities of interest (step 1)

As the original fused graph represents criminal and non-criminal entities the first step is identifying key nodes within the graph that are of interest. As outlined earlier this is done using the abstract dependent concepts of inferred or alleged predicate offences, money laundering offences, associated offences, proceeds realisation, and the vehicles used to perpetrate these acts (see Fig. 1).

So, an entity known, inferred or alleged to be involved in a criminal act is labelled "Predicate offence" (i.e. "PredO")—based on previous convictions or allegations (e.g. a person with a conviction for drug trafficking). The second class of node is the class of node involved in offences directly associated to predicate offences (i.e. "AssocPredO")—for example, a person with a conviction for identity fraud. A third class of node are those nodes involved in an "Alleged money laundering offence" (i.e. "AMLO"), which is denoted as any entity that has been recorded as being directly involved in a suspicious transaction. A fourth class of node are entities that either represent a transparent domestic corporate entity (e.g. a domestic company) or are associated to a domestic corporate entity (e.g. shareholder). These nodes are labelled"Potential money laundering vehicle" (i.e. "PMLV"). The fifth class is an entity that itself, or an associate entity, demonstrates features of non-transparency (e.g. a corporate entity with shareholders based in a tax haven) is labelled as"Potential non-transparent money laundering vehicle" (i.e. "PNTMLV"). Lastly, an entity that represents an asset (e.g. property, motor vehicle), or an entity in relation to asset ownership, is labelled as a"Realised Asset" (i.e. "RA"). Using these classes of nodes we can then identify the set of "entities of interest".

This "entities of interest" subset of nodes includes all nodes labelled "Predicate offence", "Associated predicate offence", "Alleged money laundering offence" or "Potential non-transparent money laundering vehicle". Let's call these primary nodes. Nodes of the remainder label types ("Potential money laundering vehicle" and "Realised asset")—let's call these secondary nodes—are included as "entities of interest" if they are also directly connected to a primary class of node. We are only interested in the secondary class of nodes when they present proximally to nodes of the primary class. The reason for this is that secondary nodes are not of interest in isolation as they simply represent transparent businesses and assets, a ubiquitous feature of everyday activity, and so they are only relevant when coupled to the primary class of nodes.

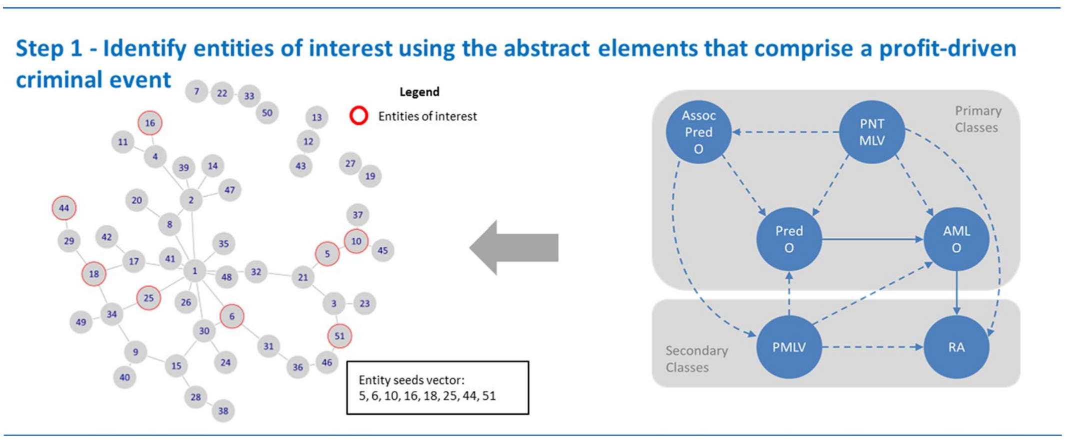


Fig. 1 This figure depicts the first step of "GraphExtract"—identifying entities of interest

For example, a secondary node may be a domestic corporate entity that has a convicted drug trafficker as a shareholder. The mechanism to identify proximity between secondary and primary nodes is entity resolution. When an entity resolution prediction is made between a pair of entities that include a primary node (e.g. organised crime member John DOE, DoB 1st January 1900 sourced from a 'criminal dataset') and a secondary node (e.g. John DOE, DoB 1st January 1900, is the shareholder of an domestic company) then the secondary node is included as an"entity of interest".

Generating the set of "entities of interest" in this fashion creates the ability to extract subgraphs at the most atomic level using a pattern that is flexible enough to deal with missing elements as expected when applying the open world assumption. Furthermore, this approach provides the basis for a generic subgraph typology.

"GraphExtract": partition the entities of interest and include mediating entities (step 2)

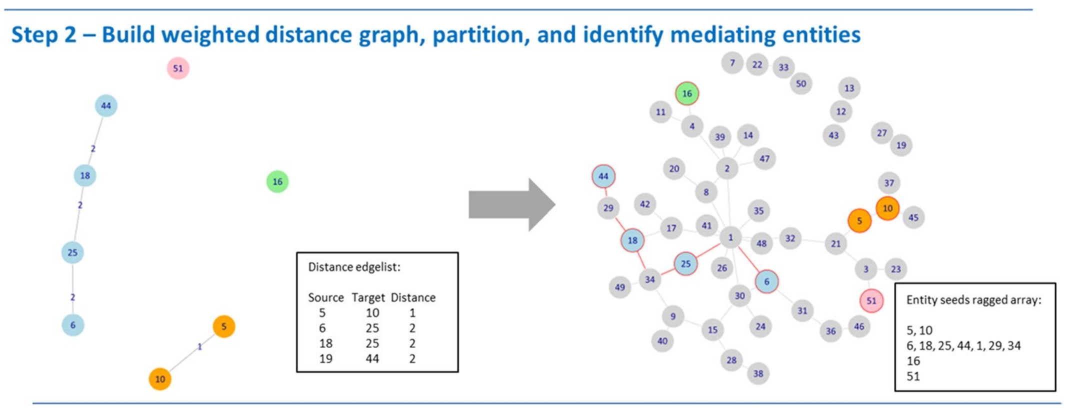
The second step is focused on partitioning all of the "entities of interest" across the entire graph, as identified in step one, leaving the residual nodes that are not "entities of interest" classless (i.e. "NA"), and then adding mediating nodes to form a ragged array of entity seeds (see Fig. 2).

This is done in five parts. First, the pairwise distance (i.e. the number of relationships or 'hops' it takes to travel between a pair of nodes) is measured between the "entities of interest". Second, this set of pairwise distances is then used to create a weighted graph. A key parameter here is the maximum graph distance allowable for consideration. The default used is 2, however a graph distance of 3 or 4 is reasonable and is totally dependent on the data and how it is modelled. Third, community detection is undertaken on the weighted graph to partition the "entities of interest". Communities are defined here as subsets of nodes that have a denser set of intra-connections relative to their inter-connections with nodes of other communities. Fourth, the partitioning is applied to the original fused graph (Fig. 2 illustrates this as the colouring of nodes, with classless nodes coloured grey). Fifth, local mediating nodes are identified for each "entities of interest" partition and added to form a ragged array of entity seeds.

Due to the high level of uncertainty present the goal of step two is focused on identifying groups of actors that are proximal to one or more criminal events and yet distant from other criminal events. Criminal actors are dynamic and over time will generally form a range of functional criminally focussed relationships [39–41] so a range of functional groups exist—from simple through to multiple overlapping functional groups.

It is important to note here that each partition of "entities of interest" is non-overlapping, so we cannot have an entity of interest that exists in two different clusters. This helps ensuring that each extracted subgraph is at its most atomic level possible and reduces the possibility of extracting compound criminal activity which can create nested structures for step three where we identify overlapping subgraphs. The selection of the most appropriate community detection algorithm to use is dependent on a number of factors including network topology, computational expense, scalability and granularity required. The InfoMap community detection algorithm [42] was implemented in this instance for reasons of computational speed and community membership accuracy and granularity. However a different application with differing circumstances may find more favourable results in a different community detection algorithm.

So now we have non-overlapping partitions or subsets of "entities of interest" that each represent different



Distance edgelist:

Source	Target	Distance
5	10	1
6	25	2
18	25	2
19	44	2

Entity seeds ragged array:

5, 10
6, 18, 25, 44, 1, 29, 34
16
51

Fig. 2 This figure depicts the second step of the process—identifying the seeds for subgraph extraction

constellations of node classes. For example, three nodes may form a subset, a person entity that is a member of a gang that has previous conviction for trafficking drugs (i.e. "PredO"), a suspicious transaction (represented as a node) of $1 million (i.e. "AMLO"), and a domestic limited company (i.e. "PMLV")—together forming a specific constellation of elements (i.e. "PredO", "AMLO", "PMLV").

As per the open-world assumption we assume that we have failed to identify all "entities of interest". This is why "local mediating nodes" are identified and added to each partition of "entities of interest". Local mediating nodes are identified by taking each "entities of interest" partition in turn and detecting all nodes that lie on the shortest path between each pair. Local mediating nodes are likely to be relevant, due to reasons of homophily and brokerage. Homophily referring to the tendency of people to associate with others similar to themselves [43], and brokerage referring to the role people play in introducing people and mediating relationships [44, 45]. For example, the entities that connect or mediate these entities of interest are likely to be relevant as they are either the vehicle of communication (e.g. phone, email) between these entities of interest, a vehicle to conduct a criminal role (e.g. business, company), a relevant geographical location (e.g. a home address), or a person entity that due to reasons of homophily and brokerage is themselves a potential person of interest but simply has not been identified as such.

At the end of step two we have the seeds required to locate the boundaries of each subgraph—the idea being that each set of seeds represents a fragment of the atomic core elements of a criminal event, and that the next steps purpose is to locate the boundaries of the subgraph within the context of the entire graph, understanding that the boundaries between of each subgraph will often overlap [1, 36]. The vectors of seed nodes are represented as a ragged array (see Fig. 3).

"GraphExtract": locate boundaries of subgraphs and generate subgraphs (step 3)

Within the third step each vector from the ragged array is then used as the start seeds to generate each induced subgraph. This step is dependent on a range of elements including the data-model, breadth, quality, completeness, and variability of the data. A basic subgraph extraction approach would entail an ego-based approach where any nodes of distance n from the seed nodes are included. This approach is a useful starting position, however the subgraph extraction algorithm implemented takes a more nuanced approach. It must be noted at this point that the algorithm developed is optimised specifically for the purposes of the case study data, and whilst fundamentally generic absolutely requires optimisation when applied to any new domain.

The algorithm takes the set of seed nodes as an input and identifies all neighbouring nodes, less supernodes (defined through domain knowledge (e.g. casinos) and graph metadata (e.g. a phone that has a high number of connections)), iteratively adding neighbour nodes to a maximum of four hops or when the subgraph exceeds a

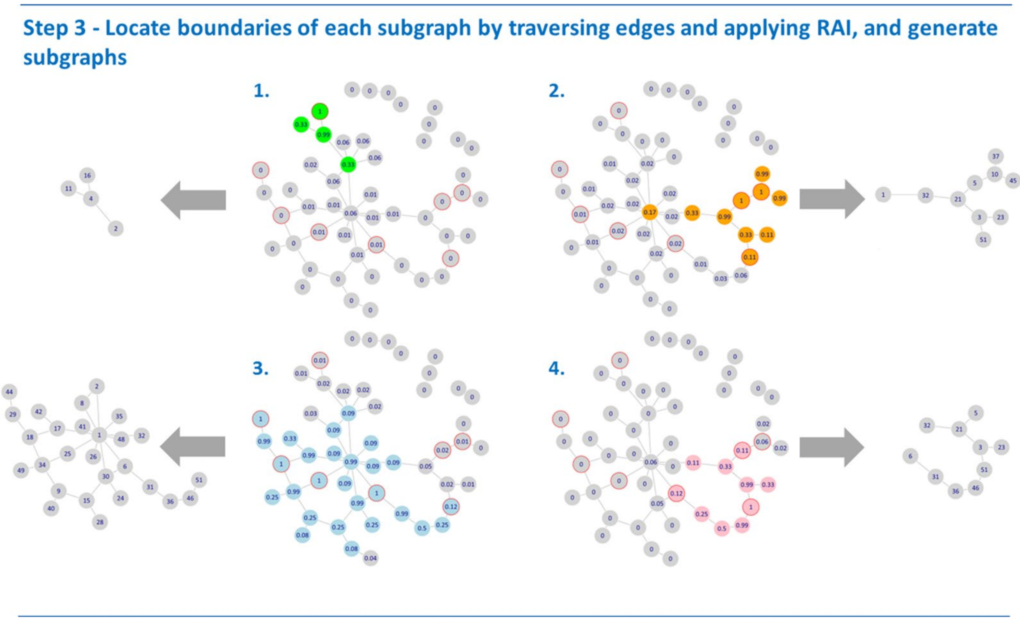

Step 3 - Locate boundaries of each subgraph by traversing edges and applying RAI, and generate subgraphs

Fig. 3 This figure depicts the third step of the process to generate subgraphs—subgraph extraction

size of one hundred and fifty nodes—the maximum subgraph size parameter.

Subsequent to the identification of this raw subgraph the Resource Allocation Index (RAI) [46] is then used to prune nodes assessed as not as relevant. The RAI works by measuring the product of the inverse normalised degree of all nodes along every shortest path between a pair. So low scores close to zero indicate a longer path marked by high degree mediating nodes and high scores close to one indicate shorter paths marked by low degree mediating nodes. RAI is applied to all pairs involving the seed nodes and is assigned to nodes on the basis of the highest RAI score for each target node. All nodes that score under a pre-defined parameter (default = 0.07) are then removed from the subgraph. The RAI is used because it algorithmically identifies those nodes that are most proximal to the seed nodes and unique in terms of their connections to the broader graph. In this way we avoid peripheral supernodes and focus on the more relevant nodes.

The approximate subgraph ceiling of 150 nodes has been determined from the underpinning evidence that the ceiling of social groups is approximately 150 [47, 48], in conjunction with visualisation performance (e.g. a rendering time of 10 s may be too long) and human performance (e.g. there may simply be too much overlaying detail for an intelligence analyst to easily consume and interpret).

This last point is critical as the goal is to generate meaningful subgraphs from the original fused graph, best representing the reality of the real-world (see Fig. 3). It is important to note that the quality and incompleteness of the data coupled with the specific intent and domain of the user, given the open world assumption, renders any more precise or specific mechanism to identify entities of interest or boundary location pointless. The goal is not necessarily to delineate the complete set of actors, who each had a role in a specific criminal act. But rather identifying the fragment necessary to make a contextual assessment of its importance. And then either dedicate intelligence/investigation resource, or not, to collect additional data in an iterative sense, constantly assessing this deployment of resource in the context of all visible criminal acts and the uncertainty bound to each instance. Thus locating the boundaries of each subgraph is a pragmatic rather than a sociological endeavour.

The goal here is to present meaningful subgraphs that are intended as providing a fragment of the real-world enabling a high-level contextual view of all subgraphs in relation. This creates the opportunity to understand the whole domain in context and make contextual decisions about how to deploy intelligence/investigations resource in the best way. Then from a microscopic perspective subgraphs can be used as a starting point in the intelligence/investigations process. Given the high uncertainty we often cannot hope to identify the 'real' boundaries of the functional group of actors—however this is defined—and often could not justify the effort expended at this point in the intelligence process.

"GraphExtract": construct a mesoscopic weighted graph using subgraph intersections (step 4)

The fourth step involves the construction of a mesoscopic perspective of the subgraphs by generating a weighted graph that represents how each subgraph's nodes overlap with one another—the mesoscopic graph. Edge weights are determined by the proportion of intersecting nodes between subgraphs, using the subgraph within each dyad with the lowest number of nodes as the denominator. This mesoscopic perspective is critical to understand the role each criminal subgraph has across the entire complex criminal network. It also gives the opportunity to go beyond simple aggregation based analysis on the entity and analyse emergent properties at the level of the group, in an attempt to understand group roles and how this influences prioritization (see Fig. 4) [1, 40]. Creating this contextual mesoscopic perspective produces the opportunity to utilise a range of knowledge discovery modelling approaches (see "Related work") in conjunction with domain context from which to understand how these atomic criminal subgraphs inter-relate.

At this point users can exploit the three data representations; the original fused graph, each subgraph (the microscopic view), and the mesoscopic graph (i.e. graph representing how criminal subgraphs are connected) to

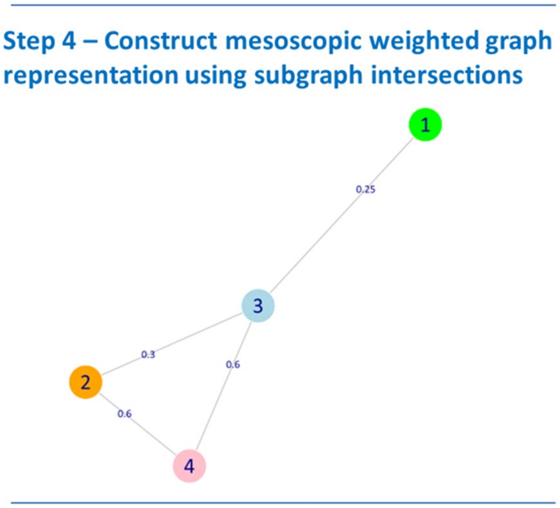

Step 4 – Construct mesoscopic weighted graph representation using subgraph intersections
1
0.25
3
0.3
0.6
2
0.6
4

Fig. 4 This figure depicts the fourth step of the process—generate the mesoscopic graph

better understand their domain through applying expert knowledge and knowledge discovery methods. Specifically the subgraphs can serve as a battery of intelligence or investigations leads, which can be presented to investigations and intelligence functions for risk mitigation.

Knowledge generation is completely open to context, purpose and interpretation however the literature covered in "Related work" section and the case study below give a firm basis on understanding how to discover knowledge from a graph perspective within this domain. Of particular importance to law enforcement and intelligence agencies, from a complex systems view, is the concept of topological vulnerability. Topological vulnerability (or attack vulnerability) refers to assessing each node for their importance in preserving network performance when under threat of removal [49, 50]. Topological vulnerability is associated to notions of network resilience, robustness and redundancy. The criminal domain is a special case as there is a very real trade-off between maintaining a robust network that is resilient to attack and a network that is efficient and carries few topologically redundant nodes [1, 39–41, 51]. Each criminal network is exposed to risk from authorities (and sometimes competitors) and dependent on other factors such as experience, access to resource and trust between actors will form a topology that's best fit for purpose. So, from law enforcement and intelligence agencies perspectives measuring which nodes and subgraphs are most important to remove for maximal network disruption, at a microscopic and mesoscopic level, is a useful metric.

Using latent knowledge, such as topological vulnerability, in combination with other knowledge such as brokerage, supply chain role, and entity criminal history, can then be used in combination for better decision-making. Having said this let's now look at an evaluation of the real-world application of "GraphExtract".

Evaluation of "GraphExtract" through a case study

Evaluation of performance is always difficult to assess in applied settings, and particularly when not only is the data classified but the lag in getting objective feedback, via investigations and prosecutions, can be in the one to four year bracket. The evaluation scope is constrained to the success of the "GraphExtract" algorithm to derive meaningful contextual fragments of real-world criminality in an automated, scalable and generic fashion. The value of such an approach versus the reliance on a reactive approach is well argued, so we feel the key criteria to assess is how scalable the algorithm is and whether the subgraphs, and resultant mesoscopic graph, are meaningful. Scalability is relatively objective in measurement however measuring 'meaningful' requires a subjective assessment.

The data used to assess the performance of the "GraphExtract" algorithm contains open source and closed source. Data includes a national companies register, national real estate register, criminal intelligence data, suspicious transactions, sanctions data, and Offshore Leaks database (see www.icij.org). The data has been transformed into a generic property graph and entity resolved (sampled f-measure 0.996) to form a fused graph of 10 million nodes and 50 million edges. The "GraphExtract" algorithm was implemented in the R language. The input (the fused graph) is represented as an igraph [52] object, and the output is a list of igraph objects. In this case over 20,000 subgraphs were generated, plus the mesoscopic graph, in a little over 102 min runtime. Step one through to four took approximately 2, 20, 58 and 22 min respectively. The mean time to generate a subgraph in step three was 0.1738 s.

The algorithm was deployed in the R language on a Windows 10 environment with a CPU utilising Intel Xeon @ 2.20 GHz (8 sockets) and 64 Gb RAM. An outline of the computational expense of the current implementation of step three of "GraphExtract" is provided in Fig. 5. Figure 5 illustrates a random sample of 1000 subgraphs, with the size of subgraph represented on the x (number of nodes) and y (number of edges) axes, the size of the dot representing the RAM expended, and the colour representing runtime (in seconds). Here we can see the variance of subgraph size as well as the increase in computational expense and runtime compared to the size of the subgraph. These performance metrics are not intended to reflect optimised production-ready software, but are only provided to give context to the scalability of the designed and implemented algorithm. The algorithm is designed to be implemented in a distributed context enabling future scalable implementations on big data.

Figure 6 depicts a pane of criminal subgraphs extracted from the original fused graph, highlighting what constellation of criminal event elements is represented. We will use these six subgraphs to reinforce the explanation of "GraphExtract" algorithm. Subgraph A. displays a cluster of suspicious transactions (red nodes) flowing from the source node to the target node. The target node has associations with organised crime entities (magenta nodes). Six of the organised crime entities adjacent to the target node have entity resolution predictions with entities in companies office data (grey nodes) and one of these criminal entities also has an entity resolution prediction with an entity from 'Offshore Leaks' data (cyan nodes) who is connected to three corporate entities based in a tax haven. This pattern depicts a real-world scenario where approximately $1 million has been transferred domestically to an entity that is a director of multiple domestic companies, has organised crime associations, and

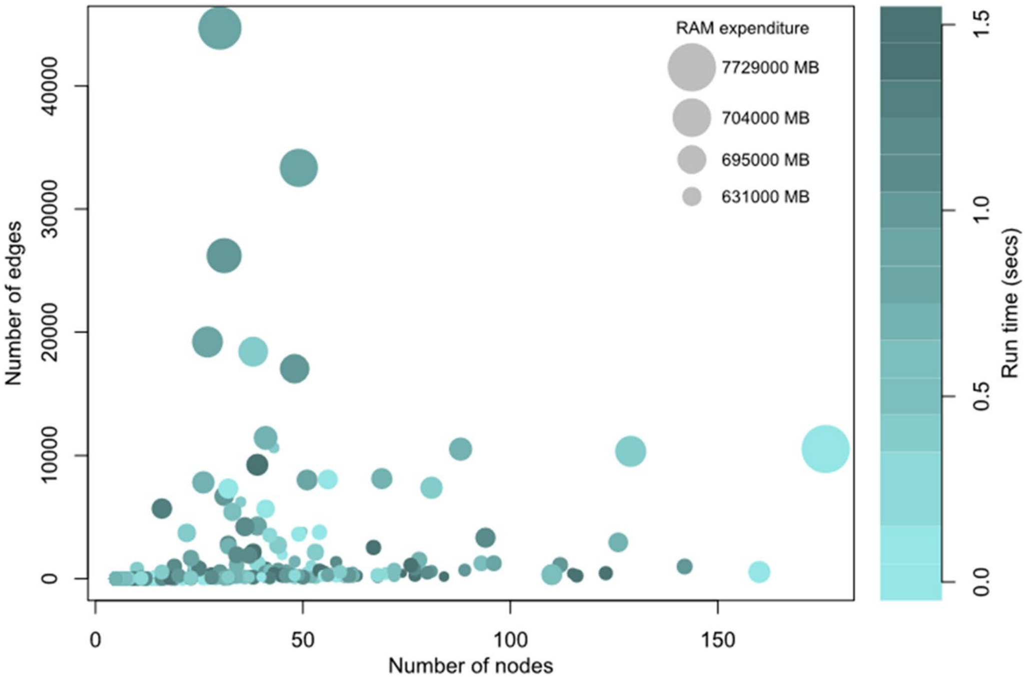


Fig. 5 This figure illustrates the computational expense of step 3 across a sample of 1000 subgraphs

is associated to offshore shell companies based in a tax haven. Subgraph A is characterised as a relevant approximately complete subgraph as it combines core elements (i.e. PredO|AMLO|PMLV|PNTMLV) of a criminal event.

Subgraph B. conveys a particular cyclic topology that is characterised as a relevant approximately complete subgraph as it combines core elements (i.e. PredO|AMLO|PMLV) of a criminal event. Again we have organised crime entities predicted to have direct involvement with corporate entities and suspicious transactions. This subgraph is a good example of the complexity of some atomic criminal events, demonstrating the non-trivial process of accurately extracting relevant complete subgraphs.

Subgraphs C and D provide examples of differing constellations but demonstrate incomplete subgraphs, either due to incomplete data, as in Subgraph C, or failure to traverse until the suspicious transaction is reached in Subgraph D.

Subgraphs E and F provide a simple demonstration of the importance of uncertainty. The entity resolution prediction (the red line) predicts that the STR entity (the pink dot) is the “same as” the domestic corporate entity (the grey dot), however there is uncertainty derived from the accuracy of the entity resolution model used. Subgraph F is a similar type of subgraph as subgraph E but the uncertainty of the entity resolution prediction is buttressed by the having the context of two proximal predictions—a demonstration of making in situ entity resolution predictions—reducing uncertainty.

Two subject matter experts (intelligence and investigations staff) were used to validate 100 of the 20,000 subgraphs in a screening exercise. The 100 subgraphs were randomly selected from a larger pool of subgraphs that contained entities involved in organised crime and significant sums of suspicious transactions. This exercise mirrors the process undertaken by law enforcement and intelligence agencies. Each expert was asked to independently rate the 100 subgraphs on their relevance, completeness, and whether they were atomic enough. The following assessments were made. 90% of the subgraphs were assessed as relevant (with 5% unsure and 5% irrelevant), 86% of the subgraphs were complete (with 2% unsure and 12% incomplete), and 88% of the subgraphs were assessed as suitably atomic (with 3% unsure and 9% too complex). The term “unsure” refers to when there was disagreement between the two experts. Irrelevance was based on a combination of entity resolution error in prediction and subgraph incompleteness. Incompleteness was based on a combination of algorithm failure and data incompleteness. The 9% of subgraphs that were not atomic enough involved a combination of data error, incomplete data, real-world complexity, and algorithm failure. False negatives were not able to be assessed due to a lack of resource.

In terms of the “GraphExtract” algorithms generic value it is of interest that the two subject matter experts

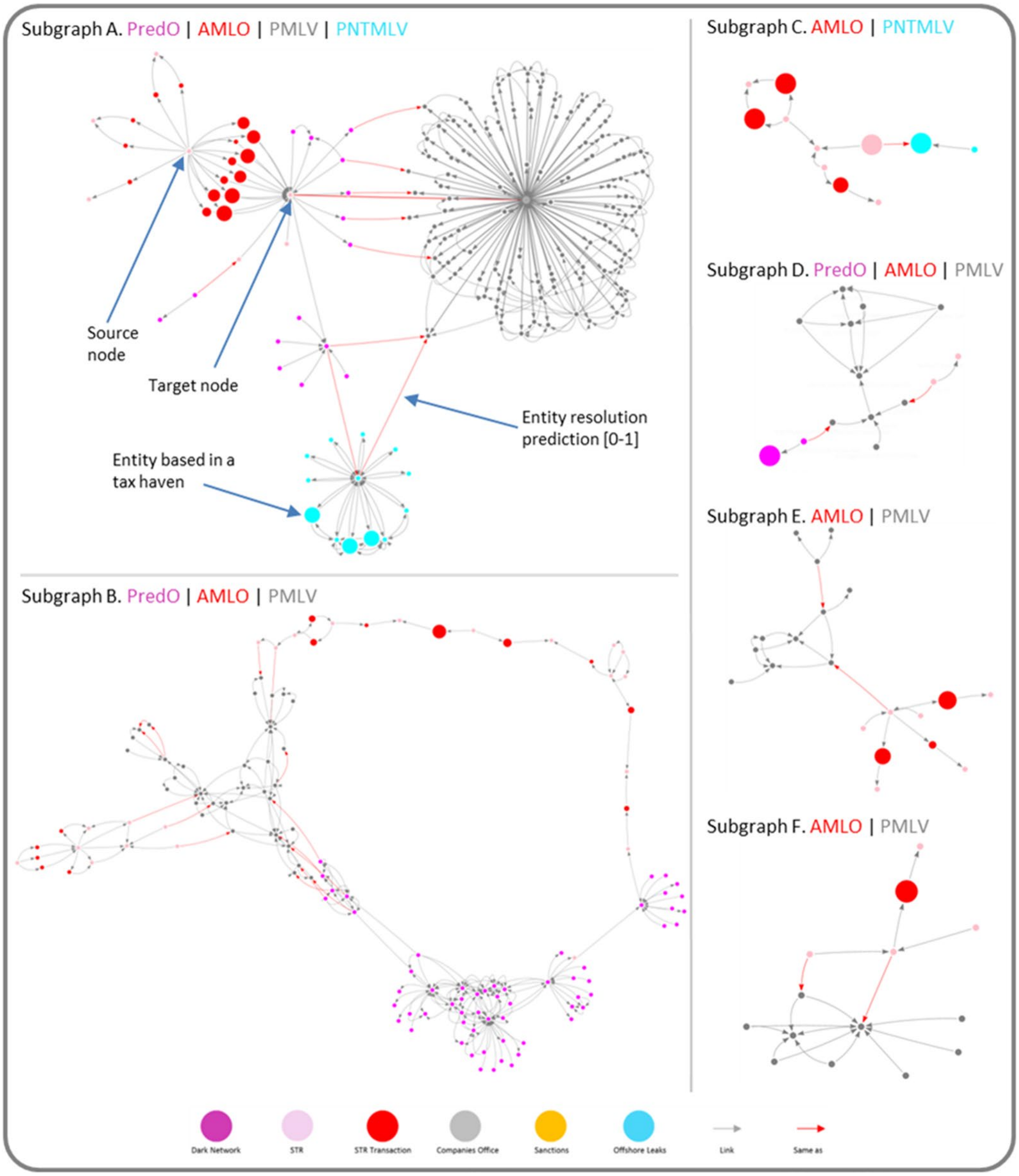

Subgraph A. PredO | AMLO | PMLV | PNTMLV
Source node
Target node
Entity resolution prediction [0-1]
Entity based in a tax haven
Subgraph B. PredO | AMLO | PMLV
Subgraph C. AMLO | PNTMLV
Subgraph D. PredO | AMLO | PMLV
Subgraph E. AMLO | PMLV
Subgraph F. AMLO | PMLV
Dark Network
STR
STR Transaction
Companies Office
Sanctions
Offshore Leaks
Link
Same as

Fig. 6 This figure gives a range of visualised subgraph examples, indicating a number of different variants

performed different roles in the law enforcement agency (Intelligence Analyst and Investigator) with two completely different perspectives—one on organised crime and the other on serious financial crime—and that of these 100 subgraphs 8 were assessed as being relevant to share with appropriate law enforcement and intelligence agencies. This hints at the generic possibilities of the algorithm. Of course this evaluation is limited to one law enforcement agency and therefore the algorithms wider applicability or portability remains untested.

Another important element to evaluate is the mesoscopic graph and particularly the topology or structure of that graph, and real-world assessment of how the mesoscopic graph reflects reality. This is critical as if there was no identifiable structure or any identifiable features that reflect reality then there is little basis to assess the subgraphs as being accurate contextual representations of atomic sets of criminal actors. Figure 7 illustrates the giant component of the mesoscopic graph, with the colour of nodes representing subgraphs that contain domestic organised crime entities (magenta), transnational organised crime entities (orange), offshore suspicious transactions (yellow), domestic suspicious transactions (grey), and other (blue). Node size represents each subgraphs total dollar amount of suspicious transactions.

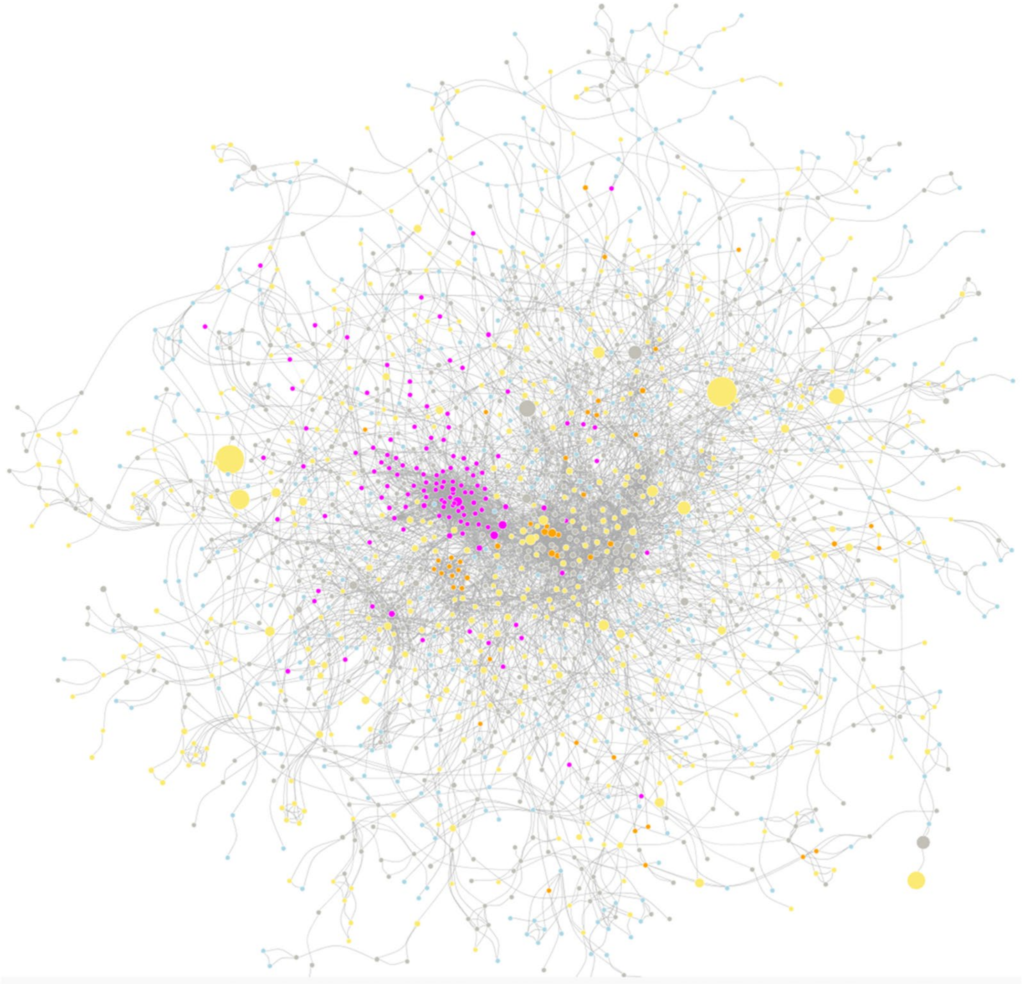


Fig. 7 This figure gives an example of the giant component of the mesoscopic graph

Blockmodelling, a well-established technique to assess structure [20] and specifically applied on criminal networks [21], was undertaken to test the structure of the mesoscopic graph. Blockmodelling detected a core/peripheral structure within the giant component, with the core representing around 10% of the subgraph nodes within the giant component. Figure 8 illustrates the core of the giant component of the mesoscopic graph divided into six distinct blocks with a seventh peripheral block (not included). Each block is represented by a different node border colour, upper right (blue), lower right (lime green), lower left (light grey border), middle left—subgraph 20848—(black), upper left (yellow), and upper centre—subgraph 5164—(magenta). Edge colour represents the proportion of intersecting nodes between subgraphs, using the subgraph within each dyad with the lowest number of nodes as the denominator. Conveying the edge colour in this way enables the reader to visually assess how connected the subgraphs are. The light grey block (lower left), containing subgraphs that all have a domestic organised crime presence, has the most overlap, however across the mesoscopic graph the mean intersection proportion between subgraphs was 0.016. The results of blockmodelling clearly indicate how the six core blocks are associated to one another in the context of the peripheral block. There is a prominent pattern of brokerage [44] with subgraph nodes 21302, 20848 and 5164 particularly prevalent, as indicated by node size. Targeting these subgraphs could potentially yield a more enduring impact impairing the efficient functioning of the entire complex system.

These findings indicate that across the entire criminal network, represented by the approximate 20,000 subgraphs within the mesoscopic graph, there is an outer periphery of 18,000 subgraphs indicating criminal activity largely unconnected to organised crime entities. Then there is a periphery of the giant component of around 1800 subgraphs which are tangentially

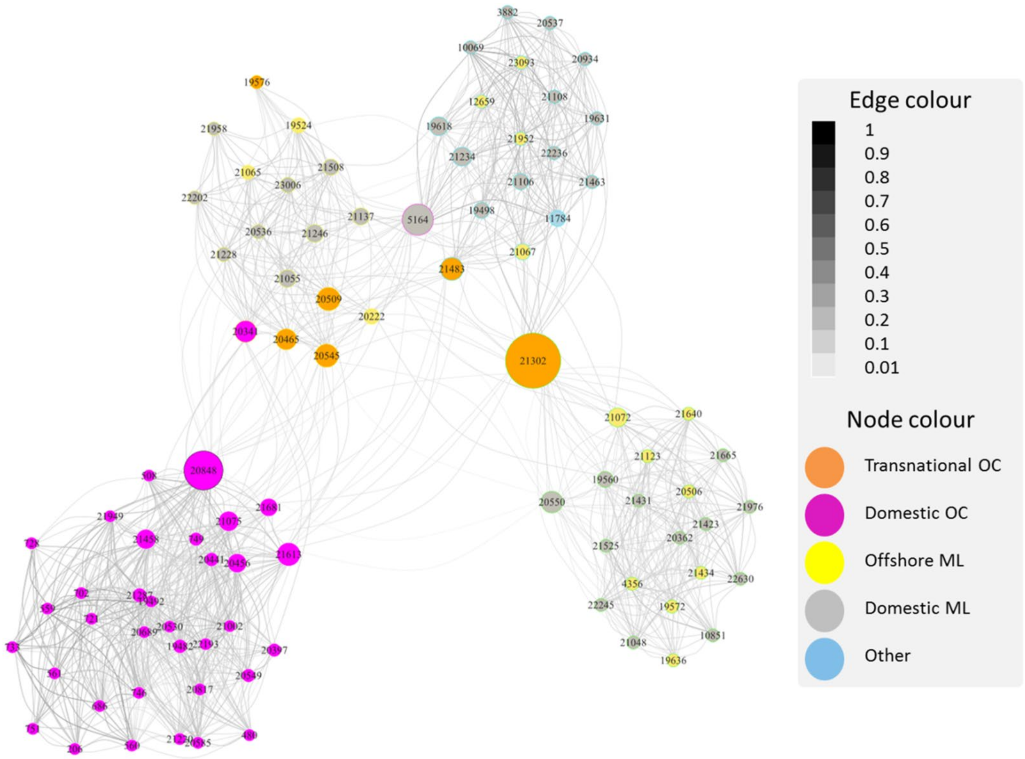


Fig. 8 This figure illustrates the six blocks (or classes) of the core of the mesoscopic graph

connected to an organised crime core of around 200 subgraphs which are brokered by a number of core subgraphs that have a strong organised crime presence.

It would be remiss if we did not at this point underscore these results with the generic caution that the data is an incomplete representation of the real-world. As such much work remains to be carried out in terms of validating and improving this approach. This does not detract from the value of the approach, but rather highlights the inherent limitations encouraging experimental frameworks to field test the value of such analytically devised approaches. Having said this the cursory findings, at both the microscopic and mesoscopic level, are enough to suggest that there is significant value in the "GraphExtract" algorithm.

Conclusion

With the advent of increased access to relevant datasets, the ability to fuse these datasets, and represent this fused data in an expressive format (as a graph), the question is then how to exploit this data asset maximally for the purpose of detecting profit-driven criminal events. As outlined earlier perspectives that view actors as sets of interconnected functional groups that act within an interconnected complex network provides an analytical position that has significant advantages over more rational actor based perspectives. Therefore, detecting criminal events and the actors that carry out these criminal events, in a contextual way that fundamentally recognises the importance of an actor's local and broader network is most logical.

The generic approach outlined above adopts this complex systems perspective generating empirically tested value at many levels, simply by attempting to understand actors in the context of their network. The first level is the detection of fragments of profit-driven criminal events, identifying functional groups of criminal actors on a prima facie basis that can be considered as proactive 'leads' for intelligence and/or investigations follow up. This demonstrates the detection dimension, with the generation of latent knowledge on each functional group providing the ability to better prioritize. The second level is the ability to view these functional groups in the context of the entire criminal system. This enables the generation of latent knowledge from a unique contextual perspective—demonstrating the ability to uncover latent knowledge to support higher level decision-making. The third level is the creation of expressive data representations that can be used as the basis to not only increase domain understanding but crucially create the opportunity for better evidence-based decision-making. Importantly any solution that ties decision-making at the micro level through to the macro level creates a connected chain of evidence base ensuring coherent aligned collective decision-making. Decision-making from strategy and strategic statements (e.g. such as maximally impairing criminal networks in a sustainable fashion) right through to how an agency decides to deploy resource (e.g. data collection on certain groups, and investigation on vulnerable but influential groups), and how intervention is operationally conducted (e.g. focus on the actors that possess unique skill sets and conduct core roles such as brokers or accountants).

Testing this approach to date has been conducted within one law enforcement agency on a fused graph of 10 million nodes and 50 million edges generating 20,000 subgraphs and a mesoscopic graph. The micro level was tested with two subject matter experts from the intelligence and investigations functions conveying a high degree of favourability over 100 sampled subgraphs with 90% of subgraphs presented deemed as relevant. However we will need to wait at least a year or two before we can empirically validate this anecdotal success due to the pace of the investigations and judicial system. At the meso level clear core-periphery structure was identified with key brokering subgraphs identified, successfully reflecting the expectations of subject matter experts. So whilst we have early indications of success broader multi-domain real-world testing of this approach needs to be carried out before we can comprehensively evaluate the value derived for law enforcement and intelligence agencies. Testing also hints at the computational efficiency and potential scalability of the algorithm, with a total run time of 102 min to generate 20,000 subgraphs and a mesoscopic graph.

A number of possible extensions have been identified. These include adoption of a node classification approach to buttress the current approach and improve the identification of "entities of interest" (step one of the "GraphExtract" algorithm), improving the subgraph extraction process (step four of the "GraphExtract" algorithm) to extract subgraphs so it is more nuanced (e.g. use topology and temporality metrics), and undertaking testing on varying datasets and contexts with differing user requirements (e.g. a national intelligence perspective) to ensure wide applicability.

Authors' contributions

DR was the Principal Investigator of this work. He is solely responsible for the development of the approach outlined, its evaluation, and writing the manuscript. CS provided feedback for manuscript improvements. All authors read and approved the final manuscript.

Acknowledgements

DR is indebted to Dr. Paulo Gottgtroy for his continued support and mentorship; and to the New Zealand government for the opportunity to evaluate the approach.

Competing interests

Not applicable.

Availability of data and materials

Not applicable.

Consent for publication

Not applicable.

Ethics approval and consent to participate

Not applicable.

Funding

Not applicable.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 2 March 2018 Accepted: 12 August 2018
Published online: 17 August 2018

References

1. Robinson D, Scogings C (2017) Picking high-level fruit in dark trees: using complex systems analytics to detect and understand crime. In: Colarik A, Jang-Jaccard J, Mathrani A (eds) Cyber security and policy: a substantive dialogue. Massey University Press, Auckland, pp 87–108
2. Harper WR, Harris DH (1975) The application of link analysis to police intelligence. Hum Factors 17(2):157–164
3. Schroeder J, Xu J, Chen H, Chau M (2007) Automated criminal link analysis based on domain knowledge. J Am Soc Inf Sci 58(6):842–855
4. Maeno Y (2007) Node discovery problem for a social network. arXiv:07104975. http://arxiv.org/abs/0710.4975. Accessed 10 Jun 2015
5. Turner JC (1991) Social influence. Thomson Brooks/Cole Publishing Co. http://doi.apa.org/psycinfo/1992-97487-000. Accessed 19 Apr 2015
6. Sparrow M (1991) The application of network analysis to criminal intelligence: an assessment of the prospects. Soc Netw 13(3):251–274
7. McGloin JM, Nguyen H (2013) The importance of studying co-offending networks for criminological theory and policy. In: Morselli C (ed) Crime and networks. Taylor and Francis, United States, pp 13–27. https://doi.org/10.4324/9781315885018
8. Klerks P (2001) The network paradigm applied to criminal organizations: theoretical nitpicking or a relevant doctrine for investigators? Recent developments in the Netherlands. Connections 24(3):53–65
9. Coles N (2001) It's not what you know—it's who you know that counts. analysing serious crime groups as social networks. Br J Criminol 41(4):580–594
10. Morselli C (2005) Contacts, opportunities, and criminal enterprise. University of Toronto Press, Toronto
11. Malm A, Bichler G, Nash R (2011) Co-offending between criminal enterprise groups. Global Crime 12(2):112–128
12. Bondy JA, Murty USR (2008) Graph theory. Springer, New York, p 651
13. Benjelloun O, Garcia-Molina H, Menestrina D, Su Q, Whang SE, Widom J (2009) Swoosh: a generic approach to entity resolution. VLDB J 18(1):255–276
14. Robinson D (2016) The use of reference graphs in the entity resolution of criminal networks. In: Chau M, Wang GA, Chen H (eds) Intelligence and security informatics. Springer, Berlin (cited 2016 May 24), pp 3–18 (Lecture Notes in Computer Science). https://doi.org/10.1007/978-3-319-31863-9_1
15. Xu J, Chen H (2008) The topology of dark networks. Commun ACM 51(10):58
16. Hu D, Kaza S, Chen H (2009) Identifying significant facilitators of dark network evolution. J Am Soc Inform Sci Technol 60(4):655–665
17. Junttila T, Kaski P (2007) Engineering an efficient canonical labeling tool for large and sparse graphs. In: 2007 proceedings of the ninth workshop on algorithm engineering and experiments (ALENEX). Society for

Industrial and Applied Mathematics, pp 135–49 (Proceedings). http://epubs.siam.org/doi/abs/10.1137/1.9781611972870.13

18. Prado A, Plantevit M, Robardet C, Boulicaut JF (2013) Mining graph topological patterns: finding covariations among vertex descriptors. IEEE Trans Knowl Data Eng 25(9):2090–2104
19. Palla G, Derényi I, Farkas I, Vicsek T (2005) Uncovering the overlapping community structure of complex networks in nature and society. Nature 435(7043):814–818
20. Wasserman S, Faust K (1994) Social network analysis: methods and applications. Cambridge University Press, Cambridge
21. Xu J, Chen H (2005) Criminal network analysis and visualization. Commun ACM 48(6):100–107
22. Chen H, Chung W, Xu JJ, Wang G, Qin Y, Chau M (2004) Crime data mining: a general framework and some examples. Computer 37(4):50–56
23. Xu J, Chen H (2003) Untangling criminal networks: a case study. In: Chen H, Miranda R, Zeng DD, Demchak C, Schroeder J, Madhusudan T (eds) Intelligence and security informatics. Springer, Berlin (cited 2015 Mar 26). pp 232–248 (Lecture Notes in Computer Science). https://doi.org/10.1007/3-540-44853-5_18
24. Shakarian P, Martin M, Bertetto JA, Fischl B, Hannigan J, Hernandez G et al (2015) Criminal social network intelligence analysis with the gang software. https://asu.pure.elsevier.com/en/publications/criminal-social-network-intelligence-analysis-with-the-gang-softw. Accessed 21 Nov 2017
25. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech Theory Exp 2008(10):P10008
26. Michalak K, Korczak J (2011) Graph mining approach to suspicious transaction detection. In: 2011 Federated conference on computer science and information systems (FedCSIS), pp 69–75
27. Mookiah L, Eberle W, Holder L (2014) Detecting suspicious behavior using a graph-based approach. Visual analytics science and technology (VAST), 2014 IEEE Conference, pp 357–358
28. Huang D, Mu D, Yang L, Cai X (2018) CoDetect: financial fraud detection with anomaly feature detection. IEEE Access 6:19161–19174
29. Shaikh MA, Wang J, Yang Z, Song Y (2007) Graph structural mining in terrorist networks. In: Advanced data mining and applications. Springer, Berlin, pp 570–577 (Lecture notes in computer science). https://doi.org/10.1007/978-3-540-73871-8_54. Accessed 29 Oct 29
30. Ozgul F, Erdem Z, Bowerman C, Bondy J (2010) Combined detection model for criminal network detection. In: Chen H, Chau M, Li S, Urs S, Srinivasa S, Wang GA (eds) Intelligence and security informatics. Lecture notes in computer science. Springer, Berlin, pp. 1–14. https://doi.org/10.1007/978-3-642-13601-6_1
31. Wang T, Rudin C, Wagner D, Sevieri R (2015) Finding patterns with a rotten core: data mining for crime series with cores. Big Data 3(1):3–21
32. Li X, Cao X, Qiu X, Zhao J, Zheng J (2017) Intelligent anti-money laundering solution based upon novel community detection in massive transaction networks on spark. In: 2017 fifth international conference on advanced cloud and big data (CBD), pp 176–181
33. Fortunato S (2010) Community detection in graphs. Phys Rep 486(3–5):75–174
34. Carley KM, Reminga J, Kamneva N (1998) Destabilizing terrorist networks. Dynamics networks project in CASOS at CMU. http://repository.cmu.edu/cgi/viewcontent.cgi?article=1031&context=isr. Accessed 19 Apr 2015
35. Krebs VE (2002) Mapping networks of terrorist cells. Connections 24(3):43–52
36. Morselli C (2010) Assessing vulnerable and strategic positions in a criminal network. J Contemp Crim Justice 26(4):382–392
37. Everton DSF (2013) Disrupting dark networks. Cambridge University Press, New York, p 482
38. Morselli C, Grund T, Boivin R (2015) Network stability issues in a co-offending population. In: Bichler G, Malm AE (eds) Disrupting criminal networks: network analysis in crime prevention. First Forum Press, Boulder, pp 47–66
39. Carley KM (2003) Dynamic network analysis. Citeseer. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.299.8946&rep=rep1&type=pdf. Accessed 27 Jun 2015
40. Carley KM (2006) Destabilization of covert networks. Comput Math Organiz Theor 12(1):51–66
41. Morselli C (2009) Inside criminal networks. Springer, New York (Studies of organized crime; vol 8). http://link.springer.com/10.1007/978-0-387-09526-4, Accessed 27 Mar 2015
42. Rosvall M, Bergstrom CT (2008) Maps of random walks on complex networks reveal community structure. Proc Natl Acad Sci 105(4):1118–1123
43. McPherson M, Smith-Lovin L, Cook JM (2001) Birds of a feather: homophily in Social Networks. Ann Rev Sociol 27(1):415–444
44. Gould RV (1989) Fernandez RM. A formal approach to brokerage in transaction networks. Sociological methodology, Structures of mediation, pp 89–126
45. Morselli C, Roy J (2008) Brokerage qualifications in ringing operations*. Criminology 46(1):71–98
46. Zhou T, Lü L, Zhang Y-C (2009) Predicting missing links via local information. Eur Phys J B 71(4):623–630
47. Dunbar RIM (1992) Neocortex size as a constraint on group size in primates. J Hum Evol 22(6):469–493
48. Hernando A, Villuendas D, Vesperinas C, Abad M, Plastino A (2009) Unravelling the size distribution of social groups with information theory on complex networks. https://arxiv.org/abs/0905.3704. Accessed 2 Mar 2018
49. Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286(5439):509–512
50. Holme P, Kim BJ, Yoon CN, Han SK (2002) Attack vulnerability of complex networks. Phys Rev E 65(5):056109
51. Simmel G, Wolff KH (1950) The sociology of georg simmel. Vol 92892. Simon and Schuster. Accessed 22 Jun 2015
52. Csardi G, Nepusz T (2006) The igraph software package for complex network research. InterJournal Complex Syst. http://igraph.org. Accessed 6 Dec 2017