Mining complex trees for hidden fruit: A graph–based computational solution to detect latent criminal networks.

A thesis presented in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

in

Information Technology

at

Massey University, Albany, New Zealand

David Robinson

2019

Abstract

The detection of crime is a complex and difficult endeavour. Public and private organisations – focusing on law enforcement, intelligence, and compliance – commonly apply the rational isolated actor approach premised on observability and materiality. This is manifested largely as conducting entity-level risk management sourcing 'leads' from reactive covert human intelligence sources and/or proactive sources by applying simple rules-based models. Focusing on discrete observable and material actors simply ignores that criminal activity exists within a complex system deriving its fundamental structural fabric from the complex interactions between actors - with those most unobservable likely to be both criminally proficient and influential. The graph-based computational solution developed to detect latent criminal networks is a response to the inadequacy of the rational isolated actor approach that ignores the connectedness and complexity of criminality.

The core computational solution, written in the R language, consists of novel entity resolution, link discovery, and knowledge discovery technology. Entity resolution enables the fusion of multiple datasets with high accuracy (mean F-measure of 0.986 versus competitors 0.872), generating a graph-based expressive view of the problem. Link discovery is comprised of link prediction and link inference, enabling the high-performance detection (accuracy of ~0.8 versus relevant published models ~0.45) of unobserved relationships such as identity fraud. Knowledge discovery uses the fused graph generated and applies the "GraphExtract" algorithm to create a set of subgraphs representing latent functional criminal groups, and a mesoscopic graph representing how this set of criminal groups are interconnected. Latent knowledge is generated from a range of metrics including the "Super-broker" metric and attitude prediction.

The computational solution has been evaluated on a range of datasets that mimic an applied setting, demonstrating a scalable (tested on ~18 million node graphs) and performant (~33 hours runtime on a non-distributed platform) solution that successfully detects relevant latent functional criminal groups in around 90% of cases sampled and enables the contextual understanding of the broader criminal system through the mesoscopic graph and associated metadata. The augmented data assets generated provide a multi-perspective systems view of criminal activity that enable advanced informed decision making across the microscopic mesoscopic macroscopic spectrum.

Preface

The ethical considerations of this research were explored with the Massey University ethics committee with the details of that advice contained within Appendix B. Further ethics consideration was not sought as no personal information was disclosed that would enable the identification of a legal entity (person or corporate). Refer to 3.1 Evaluation Methodology on page 54 for details of how the ethical risk was mitigated, and the protocols in relation to the handling of the data.

Publications by the author that are relevant to this thesis include:

1. Robinson, D. (2016). The Use of Reference Graphs in the Entity Resolution of Criminal Networks. In M. Chau, G. A. Wang, & H. Chen (Eds.), PAISI 2016. LNCS, 9650 (pp. 3-18). Springer, Cham. https://doi.org/10.1007/978-3-319-31863-9_1

2. Robinson, D., & Scogings, C. (2017). Picking High Level Fruit in Dark Trees: Using Complex Systems Analytics to Detect and Understand Crime. In A. Colarik, J. Jang-Jaccard, & A. Mathrani (Eds.), Cyber Security and Policy: A substantive dialogue (pp. 87-108). Auckland: Massey University Press.

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.

3. Robinson, D., & Scogings, C. (2018). The detection of criminal groups in real-world fused data: using the graph-mining algorithm "GraphExtract". Security Informatics, 7(2), 1. https://doi.org/10.1186/s13388-018-0031-9

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.

Contents

List of figures

List of tables

Introduction [chapter 1]

1.1 Introduction

A variety of organisations are interested in the detection of crime. These organisations range from government agencies focused on law enforcement (e.g. police, tax administration, counter-terrorism, financial crime) and regulatory/compliance (e.g. medicine manufacture) through to private organisations, such as casino's and banks, focused on fraud and money laundering detection, and a broader set of organisations focused on insider crime (e.g. internal fraud, corruption).

The detection and intervention of crime is traditionally focused on rational isolated actor approaches that focus on identifying entity-level targets that are obvious which generate "high enough value". These targets are colloquially known as "low hanging fruit". So, the rational isolated actor approach to the detection of crime is dependent on the *observability* and *materiality* (significance) of the criminal events and the actors involved in conducting those criminal events and ignores connections between actors. Whilst this targeting approach may seem reasonable and even superficially intuitive, fundamental questions have been raised about its sustainable effectiveness and efficiency.

Indeed, it is posited that those actors that are persistently most unobservable in relation to material criminal activity are likely to be the most successful and influential criminal actors. And if left to flourish, these residual "high hanging fruit", have the opportunity to mature and embed themselves within the existing criminal network. The implication is that these embedded and high performing actors then go on to provide the structural fabric for functional ego-centric neighbourhoods to evolve, and organically organise, which results in efficient access to scarce resource and propagates crime through a variety of mechanisms. This then results in a "high" performing network resilient to both endogenous and exogenous shocks.

Associated to the targeting of these "low hanging fruit" is the presence of crime mitigation strategies that are well intentioned but simply do not translate to operational activity. This decoupling of strategy and operational activity leads to unsystematic and injudicious application of strategy, with decision-making exclusively devolved from senior management decision-makers to operational business units who are often driven by expertise based reactive motivations.

The core element that drives this "low hanging fruit" approach is the absence of a system platform that firstly provides an integrated view of data, both internal and external, of the criminal problem and

secondly uses this integrated data as an asset to apply computational knowledge discovery in strategic and operational contexts, enabling quality contextual decision-making.

There are proprietary solutions available on the market targeting the crime domain that provide the first element of data visibility, through the integration of heterogeneous datasets using a generic data model allied to a presentation application (e.g. Palantir®, IBM Coplink®), however none of these solutions provide high quality entity resolution, link discovery or novel knowledge discovery that couples strategic knowledge and operational knowledge together in an integrated way.

The content covered in this paper is however such a computational solution. A solution that fuses data in a highly efficient and accurate way, maximises the identification of unobserved data, and uses this foundation to provide explicit knowledge, enabling systematic and judicious application of strategic and operational resource.

This solution, known as Graph-based Criminal Network Detection (GCND), is founded on the extensive body of research that has emerged over the last couple of decades on crime from a complex systems perspective, and indeed the broader application of over half a century of complex systems thinking, and the independently growing body of research on measuring and understanding crime from a macro, or strategic, perspective.

The body of complex systems research focusing on crime has various labels including dark networks, criminal networks and illicit networks, reflecting the different perspectives the authors have derived from including sociology, psychology, criminology, computer science, and information systems perspectives, from many applied fields including compliance, law enforcement, counter-terrorism, and military agencies. However, the common thread is the complex systems umbrella. This body of research grew from two areas of research, the body that emphasised the adoption of social network analysis (SNA) and other complex systems technologies to the criminal domain (Sparrow, 1991; McAndrew, 1999; McIllwain, 1999; Klerks, 2001; Coles, 2001; Chen, Chung, Xu, Qin, Wang & Chau, 2004; Everton, 2013), and the body that undertook empirical research applying complex systems technologies to real-world problems (Dorn, Murji & South, 1992; Carley, Lee & Krackhardt, 2001; Krebs, 2002; Xu & Chen, 2003; Morselli, 2005; Robinson & Scogings, 2017).

Independent to this is the empirically derived findings from the applied settings of compliance and criminal focussed government agencies looking to improve the way that non-compliance and crime are measured and understood for the purposes of creating strategies to mitigate these problems to a maximal extent given the finite resource available (Black & Beken, 2001; Morrison, 2002; Williams & Godson, 2002; Beken, 2004; Ratcliffe, Strang & Taylor, 2014).

These bodies of research have now evolved to provide an extensive theoretical and empirical platform from which to base the design of a computational solution that provides generalizable yet contextual insight to users - users that have an active role in the detection and intervention of real-world crime.

The specific objectives of the research were to develop a computational solution fundamentally evolving current computational approaches of crime detection. A solution founded on the paradigm of complex systems, using the R programming language, built through the creation of the following packages:

1. EntityResolution,
2. LinkDiscovery, and
3. KnowledgeDiscovery

These three packages together form an entire product that creates the ability to generate:

- A highly accurate fused view of the various input datasets,
- Identifies, with high accuracy, key unobserved relationships between entities,
- Enabling the detection of latent functional criminal groups,
- Providing a view of how each of these criminal groups are interconnected as a complex system, and
- With a range of relevant metadata enabling better informed decision-making,
- That is scalable on graphs up to 18 million nodes, and
- Fast enough for useful real-world implementation.

GCND needs to not only achieve the aim above but also, due to its modular design, create the opportunity for ongoing development in a flexible way. This extensibility allows users to deploy the generalizable modules in a more context dependent less abstract way maximising the utility of the domain context.

The evaluation of each module, its constituent components, and the system in its entirety, was explicitly measured on an empirical basis, through a combination of quantifiable data metrics and independent qualitative expert assessment, and against comparative relevant published work where possible. The data employed to evaluate the computational solution includes relevant open and closed data. Evaluation data includes Sanctions data, Offshore Leaks, New Zealand Companies Office and New Zealand criminal data.

1.2 Scope

The modules of the code will be described in detail with each component empirically tested to ensure the aim is achieved both in a modular discrete sense and in an overall aggregated sense. The scope of GNCD is outlined in figure 1.1 with green modules only included. Those elements in blue, whilst key in the real-world application, are not included within the scope of this thesis. The reason that data modelling is outside scope is because this step is totally dependent on the input data representations and not built into the generalised computational solution. Visualisation is outside scope as, whilst there are many visualised elements within GCND, a technical focus on visualisation and how it has been applied in GCND would bring no significant novel value and only serve to create unnecessary complexity taking the focus off the analytical modules.

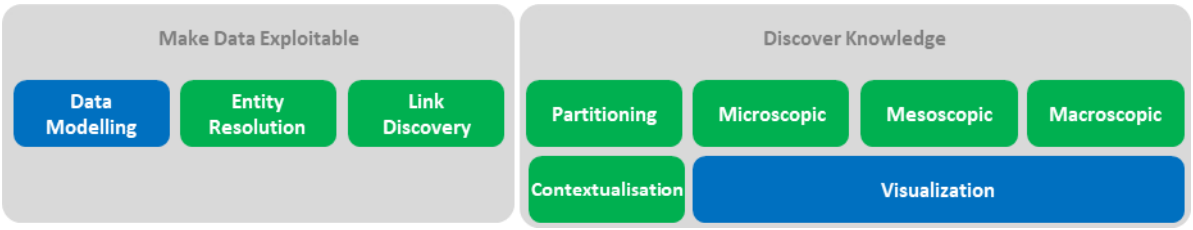


Figure 1.1. This figure outlines the modular design of GCND, with the green modules specifically within scope.

Testing and evaluation has been conducted using a range of data sets, including the Offshore Leaks, and criminal data provided from the NZ government, with the intent of testing the solution on data up to ~18 million nodes and ~93 million edges. The criminal data has been collected focusing on entities involved in the illicit drug trade and/or money laundering, and therefore the generalizability of this approach to all types of crime will remain empirically unproven. However, the majority of crime, even opportunistic and violent crime, will be influenced by the complex system but may be more complex to empirically test. The evaluation data used here creates a clear and useful, but not perfect, benchmark from which users/researchers can separately assess the value of GCND, and its constituent technologies, for their respective purposes.

The focus of this research is from an applied computational perspective, and particularly from a modelling perspective that focuses on underpinning theoretic constructs, rather than the methods or code. More specifically, the applied problem of detecting crime has been framed as a computational problem that fundamentally utilises the complex systems paradigm as a basis for the design of a solution that creates real-world value. Ultimately the first hurdle to determine the value of this solution is in its ability to provide generalizable value within the criminal domain, and assuming this is the case secondarily the novel way the technology has been designed and constructed in a way that is explicitly tied to theoretic and empirical knowledge.

Moreover, the reusability of concepts will be a common theme providing novel perspectives in how these concepts, and the metrics used to measure them, can be employed in previously unexplored ways. The modular design is intended to enable agile iterative development in a targeted way. A by-product of this modular design is the transparency and accessibility of understanding the products construction. This is a considerable benefit as the value, from a scientific progress point of view, and novelty is the contextual entirety of the system.

The interdisciplinary nature of the work undoubtedly opens this research up to extensive paradigm-centric criticism, such as how specific terms have been defined and applied, or not. However, to head most of these criticisms off at the pass we would remind the reader that this research is not for instance a sociological endeavour to infer human behaviour from a contrived sample dataset, but a real-world application of a computational solution that has to contend with the real-world and all of the baseline assumptions that that confers. So, that is the way in which this research has been presented, as a coherent logical flow following how the code has been designed. We have resisted spending an inordinate amount of time on articulating, critiquing and defending the use of specific concepts due to the breadth and complexity present, and the real risk of diluting the novel valuable content included.

Furthermore, as the code incorporates a vast range of technologies each element is not detailed to a code level. To do so would simply generate excessive content and only serve to dilute the most significant elements of this work which is the creative ways to integrate multiple technologies to generate an applied system view of criminality.

In terms of deployment the solution has been coded using the R language and designed in a modular way, utilising the following R libraries; data.table (Dowle & Srinivasan, 2019), igraph (Csardi & Nepusz, 2006), fastmatch (Urbanek, 2017), stringr (Wickham, 2019), stringdist (van der Loo, 2014), stringi (Gagolewski, 2019), kernlab (Karatzoglou, Smola, Hornik, & Zeileis, 2004), reshape2 (Wickham, 2007), doParallel (Microsoft Corporation & Weston, 2019), foreach (Microsoft Corporation & Weston, 2019), visNetwork (Almende, Thieurmel, & Robert, 2019), rmarkdown (Allaire, Xie, McPherson, Luraschi, Ushey, Atkins, Wickham, Cheng, Chang, & Iannone, 2019), knitr (Xie, 2019), rpart (Therneau, & Atkinson, 2019), fst (Klik, 2019), and parallel (R Core Team, 2019).

Like any computational solution there are inherent limitations to the product. The most obvious limitations for this solution is runtime and scalability, however the goal of the current non-engineered incarnation of the solution was tested to be performant up to ~18 million nodes and ~93 million edges in around 33 hours (with an obvious dependence on the computational platform). The performance is assessed as respectable given the decision to test the models using R, which enables a very quick

development cycle and a broad set of applicable packages from which to continue extending the current version.

1.3 Significance

The significance of this paper is manifested in four dimensions. Firstly, the most obvious outcome is the contribution to the practical mitigation of crime, via the direct application of this code on criminal problems, and the contribution of empirically tested methods, in part and in entirety, that can be applied in a conceptual manner to existing capability. Secondly, from the perspective of dark network research many areas have been empirically repeated, some ideas extended, and some new approaches developed. Thirdly, some techniques have been empirically tested in a new domain which generalises their applicability. Fourthly, some specific novel methods have been developed that can be applied beyond the domain of crime.

1.4 Organisation of chapters

This thesis consists of eight chapters divided into three parts. In Part A (A Survey of Relevant Literature) we explicitly outline the limitations of the "low hanging fruit" model which serves to indicate what elements need to be addressed to evolve to a new approach. The complex systems paradigm is introduced as the basis for a solution, which is followed by an introduction to GCND and the modules that comprise GCND to provide a contextual backbone to organise the survey of the literature [chapter 2]. The survey covers the mechanics and context of relevant dark network and broader complex systems concepts and metrics, and the criminological based research on the measurement and understanding of crime for decision-making purposes. This will be broken into the following sections; Introduction, Make Data Exploitable (Data Modelling, Entity Resolution, Link Discovery), Discover Knowledge (Partitioning, Contextualisation, Microscopic Knowledge Discovery, Mesoscopic and Macroscopic Knowledge Discovery) to enable the reader to easily understand how that research applies directly to the GCND modules. This will create a firm foundation for Part B (Methodology, Data, Design and Implementation), where we will introduce the methodology and data sets used for evaluation [chapter 3] and then go over in detail in terms of what was designed and implemented within GCND (Make Data Exploitable [chapter 4] - Entity Resolution and Link Discovery – and Discover Knowledge [chapter 5] - Partitioning, Contextualisation, Microscopic, Mesoscopic and Macroscopic Knowledge Discovery), and a contextual evaluation of how the GCND solution performed [chapter 6]. Part C will discuss potential extensions [chapter 7] and development possibilities of the work on a module by module basis and summarise [chapter 8] the computational solution in its entirety.

Part A: A Survey of Relevant Literature

Literature Survey [chapter 2]

2.1 Introduction

2.1.1 What is the rational isolated actor or "low hanging fruit" model?

The rational isolated actor or "low hanging fruit" model is the traditional approach taken by organisations with an interest in regulatory, compliance, intelligence, or law enforcement to detect risk, non-compliance, and crime. The approach often falsely adopts the closed world view focusing on finding risk (we are using risk as the umbrella term for crime, non-compliance and risk) in data, rather than taking the open world assumption and conceptualising that data as a partial representation of the real-world. This model is generally manifested within organisations as CHIS or 'modelling', largely dependent on the availability of relevant structured data:

1. CHIS: Sourcing 'leads' from covert human intelligence sources (CHIS), either anonymously or through a cultivated informant, and
2. Modelling: Applying rule-based or supervised learning on subsets of internal data to rank risk utilising internal datasets.

Adopting this dual approach leads to a focus on the detection on "low hanging fruit" – the most obvious and easily detected – failing to target the more unobservable, successful or complex risk – the "high hanging fruit".

2.1.2 Limitations of the "low hanging fruit" model

The fundamental problem is that the "low hanging fruit" model is an over-simplified approach to a problem conceptualised in an artificial and over-simplified way.

By looking for risk in data, rather than using data to construct a view of the real-world problem, practitioners are making a raft of false assumptions. These include assuming; all of the risk is present within the dataset under assessment; each entity in the data represents a single real-world entity (i.e. there are no duplicates and no fake nodes); all relevant data attributes are present to make an informed decision; the value of the dataset for identifying risk will not decay; and entities engage in criminal

activity in isolation. These assumptions made in concert result in artificially focusing on the most easily observable instances with the highest measurable materiality, and ignoring the most unobservable potentially high value targets.

At a group level "low hanging fruit" approaches tend to use simple data aggregation to identify groups of interest, again displaying a data-centric view rather than a problem-centric view. For example, group aggregation could be based on overt membership (e.g. member of Bandidos Motorcycle Club) or nationality/ethnic background (e.g. West African Organised Crime Group), failing to identify the functional group as a collection of actors engaging in a specific criminal act or acts. The simple aggregation of the obvious is not useful to detect real-world functional groups.

At the macroscopic level findings are often based on aggregates of actors and groups, ignoring emergent properties of the complex system. For example, how do groups inter-relate and what is their function in the system? What part of the supply chain do specific groups focus on or are they vertically integrated?

Furthermore, contextualisation is often a passive peripheral concept that is implicitly provided by domain experts in interpreting the findings of a model. We are using the term contextualisation here to refer to the context of the domain and the data that is available to reflect this context. Contextualisation is a critical component that is core to modelling and can serve to create opportunities to constrain the problem space and thus simultaneously simplifying and contextually embedding modelling – for both the modeller and the consumer of the output of the model. For example, explicitly acknowledging that criminality often focuses on specific commodities being traded through a supply chain is a central tenet. If we have a good idea that a class of entities are involved in the trafficking of a commodity and they are dependent on entities in the wholesaling class to generate profit then we have successfully constrained the problem space through contextualisation and can focus attention on shortest paths between entities within the trafficking class and entities within the wholesale class. The domain context is the supply chain of illicit commodities and the data that reflects this context includes the roles that a range of entities have performed across this supply chain, hence creating additional inferred knowledge about how actors that have a role in one element of the supply chain have a dependence on actors in other roles.

Similar to contextualisation is the coupling of models to the construct they are intended to measure, firmly planting the metric in the context of the problem and enabling any empirical foundation of the coupled concept-metric to be made relevant to crime. For example, the identification of relationships that uniquely connect two communities could be considered "weak ties" and therefore important to access scarce resource and knowledge (Granovetter, 1973). This creates the context to understand not

only the potential roles of entities, but which entities are critical to maintaining the structural fabric of crime.

Temporality is another concept that is often neglected for the sake of simplicity. Ignoring temporality can both introduce unacceptable error in over-simplified models and induce the overlooking of key temporal based constructs. For example, the life cycle of an entities gang membership is critical to understand because if the government is truly going to be proactive then measures should be taken to identify young people at risk of entering criminal groups, and devising intervention strategies to deter this occurring.

From a value perspective the actual enduring impact of this over-simplified approach is problematic to measure, as the problem has been artificially defined as being contained within a single dataset. The "low hanging fruit" model may generate superficially acceptable metrics, however, these metrics, upon examination, often are not related to enduring real-world impact but contrived metrics based on the source data. For example, detecting a propagating tax fraud scheme that has an exponential take up within the community at a late stage could result in millions of dollars in tax discrepancies and a handful of successful prosecutions. Whereas the detection and intervention of the same tax fraud scheme at an early stage would result in only a fraction of tax discrepancies, but a far better real-world result for all stakeholders.

2.1.3 Required elements to evolve beyond the "low hanging fruit" model

It is clear from the preceding critique that the current "low hanging fruit" approach fails to take into consideration a problem space that is characterised by complex inter-related facets between entities and the collective network of entities, the emergent properties that manifest from this interaction, the significant degree of randomness that is present in the system, the high degree of self-organising, and the broader context in which the problem space is nested. The aspects explicitly listed above together are characteristics of complex systems.

So, in response to these criticisms it is critical to develop a crime detection approach that focuses on the problem, utilises multiple datasets that, when fused, construct a reasonable representation of the key concepts of that problem, explicitly acknowledges the uncertainty of the data and takes steps to reduce that uncertainty (e.g. link prediction), targets the non-obvious and obvious targets and also takes into consideration the contextual complex system that crime is embedded within.

The key elements to evolve beyond the "low hanging fruit" approach include:

1. Understanding the problem sufficiently to then guide the minimal collection of datasets required to model this real-world problem.
2. Modelling the data available in the most explicit representation that enables the application of complex systems perspectives – a graph.
3. Make the data exploitable, by harmonising the data into a generic data model, measuring data incompleteness and error, and using technology to enhance the data as much as possible to mimic the real-world. The two key technologies here, which have been developed as modules within GCND, are entity resolution and link prediction.
4. Generate contextually useful metrics. These metrics include both pure graph theory and domain context metrics that together create the basis to better understand risk.
5. Utilise all the building block metrics derived and represented in the graph enabling the detection of criminality or risk in an abstract unsupervised way leading to newly discovered knowledge at the microscopic, mesoscopic and macroscopic levels.

The idea is that with the right problem focus we can then represent data that appropriately reflects that problem in an exploitable format – a graph. From this point the data needs to be improved so it as closely reflects the real-world representation it is aiming for, which enables a range of relevant metrics to be generated. These metrics form the building blocks from which a contextualised complex systems perspective can be applied. With the end goal of discovering knowledge across differing perspectives enabling the targeting of more complex and more important crime, and applying intervention strategies that generate a more enduring impact.

Let's now look at each of these elements in turn within the context of the literature to generate a foundation for understanding why a computational approach comprised of these elements is a wise choice. As the focus here is applied the attention will principally be on the elements that relate to the modules of the computational model, with supporting concepts covered for necessary context.

Problem-focussed rather than data-focussed

Before we jump ahead to trying to find risk in data we need to ensure that the data collected at least minimally reflects the real-world problem we are focussing on. This may seem obvious but is so regularly neglected. The computational approach that is the focus of this work is designed specifically to generate knowledge related to risk, criminality, and non-compliance so the input data needs to reflect this fact. For example, if the focus is looking at tax non-compliance then it is reasonable to expect datasets that reflect the key abstract features of this problem. These abstract features could be assets, corporate entities, non-transparency, transactions, criminality, and taxation. Mapping these abstract features to real datasets could mean the collection of data that encompasses; real property, corporate entity registrations, offshore leaks data, suspicious transactions, criminal data,

and tax administration data. Relying solely on one set of data will not be sufficient to generate knowledge about the problem. The requirement to fuse multiple datasets into one exploitable and explicit dataset creates a dependency on representing the data in the best generic data representation – a graph – and the mechanics of fusing the data into that target data representation – entity resolution.

Crime is embedded within a Complex System

The computational solution (GCND) detailed in this paper is designed to address the failing of the "low hanging fruit" model and emerges from the complex systems paradigm. The notion of complex systems is premised on a diverse body of work and the key relevant features, along with a range of other disciplines, will be introduced below in line with the modular design of the GCND (see Figure 2.1.) to ensure that the theoretical and empirical basis of the solution is explicitly tied to what has been designed.

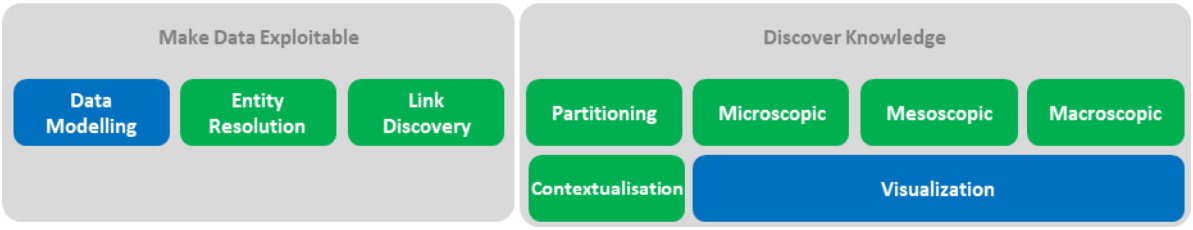


Figure 2.1. This figure outlines the modular design of GCND, with the green modules specifically covered here.

The designed modules are clustered under the two sections; "Make Data Exploitable" (chapter 4) and "Discover Knowledge" (chapter 5), which are explicit labels for what the solution is designed to achieve. The focus of this paper is on the green modules which will be extensively detailed, however the two modules in blue whilst important in their own right are not covered in any detail due to their lack of novelty and automation.

The focal point is developing a contextual system-based offender-centric perspective, removing the artificial boundaries of specific academic disciplines and starting to develop a holistic view of the problem. Understanding the problem from a range of perspectives and embedding these perspectives in the context of the problem unlocks the potential for a range of data sources, including expert views, contributed from a range of sources (e.g. police, border security, tax administration, corrections), and a range of constructs (e.g. sociology, social psychology, economics, epidemiology, genomics) from which to build generalizable models, and critically creates well-rounded relevant outputs with high face validity. In response to this view GCND has been designed from a generic complex systems basis with modular construction so a wide range of functions can be "slotted" in and run in competition.

2.2 Make Data Exploitable

The first section of the computational model covers the collective steps required to represent the relevant data that is available in a way that enables the knowledge discovery phase to maximally exploit. The modules within these two sections are; data modelling, entity resolution, and link prediction (see Figure 2.2., Figure 2.3. and Figure 2.4.). These modules collectively cover data modelling, cleansing, inference, harmonisation, transformation, integration, entity resolution, and link prediction. This section is represented as linear however elements of these modules, particularly the representation of data in a variety of devolved alternate formats, are present throughout the entire computational process. The computational model requires a generic property graph input which will be outlined within the data modelling phase, however an emphasis will not be placed on data extraction as this stage is assumed to have been performed.

2.2.1 Data Modelling

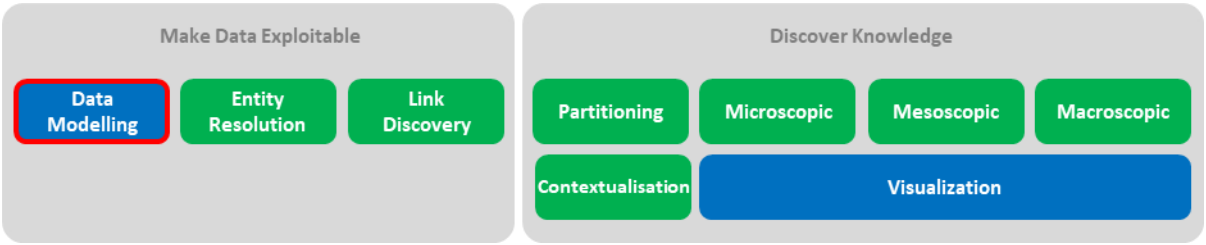


Figure 2.2. This figure outlines the modular design of GCND, with the current focus on Data Modelling.

The core data representation is a graph - specifically a hybrid property-graph representation, with semantic features and schemaless attributes. Graphs are a remarkably flexible and intuitive way to represent things. Graphs are constructed through the introduction of a set of vertices and a set of edges that connect vertices ($G=(V,E)$). Vertices are also known as nodes or entities, dependent on your background. Edges are also known as ties, arcs, lines and relationships. Property graphs also allow attributes. Within a property-graph the set of nodes have a unique identifier, and any number of attributes (e.g. label, type), and the set of edges has a unique identifier, a source node unique identifier, a target node unique identifier, and any number of attributes (e.g. type, date, source), and additionally the graph itself can have attributes. The complexity of graphs quickly expand from simple graphs that have undirected edges and no loops (reflexive edges), through to non-simple graphs that are directed (digraphs), weighted (or valued), and multiplex, through to hypergraphs that represent affiliation. Bipartite graphs (or bigraphs) are graphs where the set of nodes can be separated into two disjoint sets (U and V), with all nodes in U having at least one edge to a node in V. Bipartite

graphs are often used to portray persons overlapping membership or attendance at an event (see Wasserman & Faust, 1994).

Graphs can also be expressed as a matrix. An adjacency matrix (or sociomatrix) is an *n x n* matrix with cells representing whether node *i* is adjacent, or connected, to node *j*. Undirected graphs are represented by symmetric matrices where the upper and the lower portions of the matrix are symmetric, and directed graphs are represented by asymmetric matrices. Loops are represented by the diagonal and are empty if loops are prohibited. The matrix cells can merely reflect a binary relationship (the relationship exists or not) or can be weighted by being represented by a number. An incidence matrix is an *n x m* matrix, where *n* is the number of nodes and *m* is the number of edges, with the cells typically representing zero or one, if the edge is incident or not respectively (see Wasserman & Faust, 1994). Matrices, however have some key limitations, namely that the matrix does not allow for the representation of multiple attributes of either node nor edge, and that as graphs tend to be sparse the matrices used to represent them are computationally inefficient in terms of storage.

However, the strict property-graph format can also suffer from the sparse representation problem when a particular node attribute is not bound by a set number of fields. For example, it is common for people to have an alias, or even multiple aliases. To represent an alias as a set of person attributes is possible, and even optimal, when the graph only includes people with no more than one alias, but as soon as a person has two or even three aliases then the data model has to allow for this, introducing inefficiency into the data model. An explicit way to deal with this is to simply model aliases as a node; however this introduces greater complexity into the model. An alternative approach to deal with this problem is representing sparse attributes within a single schemaless attribute. For example, rather than representing vertex attributes as separate attributes (e.g. Label, Name, Date of Birth, Nicknames, ID, Gang, and Position) noting that very few entities may have Nicknames, ID, Gang, or Position these attributes can be represented as one generic attribute using specific delimiters to indicate the attribute type (e.g. the "Text" attribute may contain an entry like "NICKNAME; Dopey, Rico: ID; AB123456: GANG; ABC Motorcycle Gang:").

A third class of graph representation is as an edge list. Edge lists can come in the form of pairs (*i, j*) or triples (*i, j, w*) where an attribute such as weight can be included. A semantic triple is a special case of triple used in the Resource Description Framework (RDF) and is the most atomic data entity in a semantic graph. A semantic triple is constructed of a subject, predicate, and object. An adjacency list is an alternate version of an edge list which is arranged as a ragged array, where the first element of each row represents the source node and remainder elements are the relevant target nodes.

Why a graph?

Representing the data as a graph creates a number of advantages and associated disadvantages. A graph representation is a native representation of anything that includes relationships between elements. As such a graph is the perfect representation for data that represents a system view of a problem. Something the Entity Relational Model (ERM) (Chen, 1976), most often deployed as data warehouses, is not designed to do in an efficient or intuitive way.

A central by-product of this decision is that the fusion of heterogeneous datasets is made easier as dyadic and more complex relationship patterns, natively represented in a graph, can be employed more easily using graph-based entity resolution models. This removes any dependency on the use of primary keys, thus avoiding the heterogeneous database join problem (Thuraisingham, 2003), often encountered in the fusion of datasets from disparate sources. The heterogeneous database join problem is based around the dependence on the key, which can manifest as both the cost of key maintenance, the accuracy of the key and also the absence of the key which is common when integrating disparate datasets.

Having said this, computationally it can be inefficient to store or conduct certain operations on native graph objects, especially in parallel or distributed architectures which are almost exclusively table based. In these cases it is important to utilise the full range of alternate graph representations to optimise performance.

Generic graph data model

The core graph representation has a basic intrinsic model as discussed earlier – a set of nodes that has one attribute, unique identifier, and a set of edges that connect nodes that effectively have three attributes, unique identifier, source node, and target node. That model may be represented in either a graph, matrix, or table of pairs/triples. This basic graph model has been extended into a generalised generic data model that enables a wide range of graph data sets to be quickly and explicitly mapped. Additionally, it is important to create expressive labelling for both node and edge types, and explicit relationships between object types, whether that is nodes (e.g. “organisation”, “person”) or edges (e.g. “associated to”, “spouse of”), to maximise the exploitability of the data representation in a meaningful, consistent and efficient way.

2.2.2 Entity Resolution

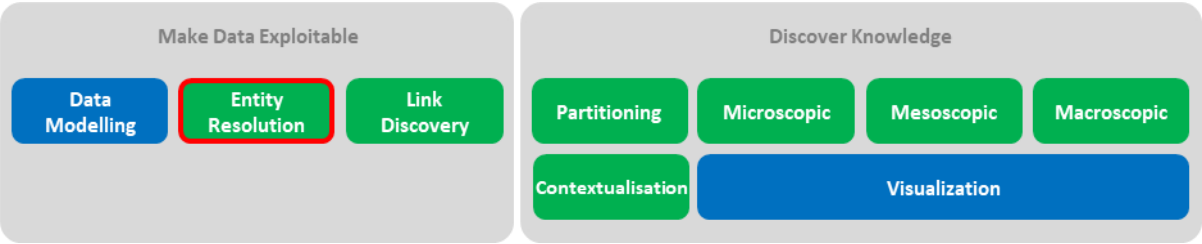


Figure 2.3. This figure outlines the modular design of GCND, with the current focus on Entity Resolution.

The need to resolve entities derives from two distinct sources; the existence of duplicates within a single dataset and the integration of heterogeneous datasets. Entity resolution (ER) is the umbrella term that encompasses identity resolution (the specific resolution of a new dataset against a reference set of known identities), data matching (a sub-component that focuses on identifying equivalent identity attributes), record linkage (the specific task of finding equivalent records within a set), deduplication (the resolving of multiple duplicate records into one), and disambiguation (the process of determining that two references are not equivalent). Entity resolution is often conducted within the broader goal of integrating data, otherwise referred to as entity-based data integration (EBDI), or data fusion. The basic process in entity resolution is to identify relevant identity attributes, cleanse and augment these attributes (e.g. split date of birth into day month and year), create a sub-set or "block" of similar entities to compare based on the identity attributes so every entity does not have to be compared with every other entity which would otherwise be intractable (e.g. only assess equivalence in entities that have the same first three letters in their family name), conduct equivalence assessment between relevant pairs of entities (e.g. using string distance metrics to measure the similarity of a name), generating an associated likelihood for each pair of entities actually representing the same real-world entity, and using the metadata generated by the equivalence assessment and likelihood generated to classify pairs of entities into categories from which to merge, link or ignore (usually defining thresholds based on a training dataset). Then the integration component of EBDI which is essentially implementing optimised entity identity management to ensure the most appropriate data is retained when contracting the vertices of the graph (Lim, Srivastava, Probhakar & Richardson, 1993), sometimes known as knowledgebase arbitration (Revesz, 1993).

At the core of entity identity management is assessing the quality of data to enable good decisions on how to make that data exploitable – particularly in entity resolution models configured to the open world assumption (Smets, 1988). Assessing the quality of data is large area of research (see Talburt, 2011); however it is clear the provenance of the data is extremely important when assessing the quality of data and making arbitration decisions. The source of data is often the first element used to

support assessment, however many attributes (identity-based attributes and other data quality metrics) and strategies can be used to inform decision-making. A further example of attributes used to assess arbitration is localised graph quality. Localised graph quality, through the assessment of topology and data quantity, can be a useful adjunct to provenance attributes as it focuses on the localised quality of data within a specific source. Common arbitration strategies include source quality, as mentioned, frequentist approaches that opt for the most common elements (e.g. choose "Andrew" over "Andrews") and data quantity (e.g. choose "Adam Bruce Smith" over "Adam B Smith"). Of course the evaluation of entity resolution, and indeed EBDI models, needs to be done to measure effectiveness. The evaluation of such models is commonly based on metrics such as the F-measure (Van Rijsbergen, 1979), and focussing on samples of negative decision pairs (Maydanchik, 2007).

Evolution of entity resolution

Dunn (1946) initially described the concept as record linkage, and this was extended into probabilistic record linkage using computers by in the 1950's and 60's (Newcombe, Kennedy, Axford, & James, 1959; Newcombe & Kennedy, 1962) using the soundex algorithm devised by Odell and Russell (1918). In 1969 Fellegi and Sunter published their model of probabilistic record linkage which has served as the basis of many applied entity resolution approaches. Then in the late 1980's and early 1990's many approximate string matching (ASM) algorithms were developed and extended, including the Jaro distance algorithm (Jaro, 1989, 1995) and its extension the Jaro-Winkler distance algorithm (Winkler, 1990). Statisticians and public health researchers then converged with computer scientists who were developing methods, such as string comparison and database sorting via attributes, to detect approximate similarity in database records (Hernández & Stolfo, 1995; Monge & Elkan, 1996). In the 2000's there was a proliferation of research on technology to extend entity resolution including data mining, machine learning, information retrieval, natural language processing, linguistics and graphs. Perhaps the most fundamental extensions over the past three decades is the focus of features or attributes to assess similarity broadening from node or content attributes (e.g. name, date of birth) to also include "context" attributes surrounding nodes such as a nodes neighbours – often manifested through transitive closure (Hernández & Stolfo, 1998). Associated to this development was the change in perspective to what is known as collective entity resolution (Bhattacharya & Getoor, 2007). Notably collective entity resolution takes a more contextual perspective than the more established pairwise approach, looking at clusters of potential matches and using transitivity (transitive closure) and exclusivity logic to support decision-making. The key areas of entity resolution research of most relevance here are the use of graph features, iterative resolution, collective entity resolution, and linguistics.

Relevant areas of entity resolution

Graph features are often neglected in table-based approaches that treat entities as isolated entities. However, viewing the problem with a systems view and representing the data as a graph provides the focus and the means to utilise graph features to improve ER performance. Multiple graph features or attributes (also referred to as context attributes) can be drawn from the graph (e.g. graph distance between a pair) to support the measurement of pair equivalence (Bhattacharya & Getoor, 2007) and other components of ER.

A category of these graph-based similarity measures is known as neighbourhood similarity. Various metrics have been developed and extended that measure this concept. Common neighbours is simply the measurement of how many neighbours *i* and *j* share, however this fails to account for the degree of connectivity that each node has. The Jaccard coefficient addresses this by taking into account the number of the union of connections the pair has, and using this as the denominator. Adamic and Adar (2003) extends this notion further by not assuming that every neighbour has equal value in determining equivalence. This idea is intuitive given that social networks generally have an approximate scale-free degree distribution, and therefore there is a higher likelihood that a pair of nodes share a relationship with a prominent node by chance when compared to the pair both sharing a relationship with a low-degree neighbour (Adamic & Adar, 2003). The Adamic-Adar similarity measure is the number of the pair's common neighbours weighted by the inverse log of the pair's degree.

Other graph concepts such as partitioning (e.g. community detection), equivalence, topology, and path-based metrics can all be utilised to augment performance in a range of ways (Bhattacharya & Getoor, 2007; Robinson, 2016).

Iterative resolution refers to employing an iterative process to a single component of the entity resolution process, like blocking (Whang, Menestrina, Koutrika, Theobald, & Garcia-Molina, 2009), or to a more substantive set of components of the entity resolution process to incrementally leverage off the evolving data representation. Graph-based entity resolution systems fundamentally provide the opportunity to exploit the evolving improvement of the data via relational aspects of the representation. The iterative conglomeration of graph nodes incrementally improves the accuracy of the graph, and hence enhances the quality of graph-based identity attributes. This data augmentation is particularly important in domains where data incompleteness, uncertainty and quality are significant limiting factors, like the criminal domain – and so the open world assumption applies (Smets, 1988).

Indeed, it is critical that any entity resolution approach explicitly acknowledges whether the domain requires an open or closed world assumption (sometimes referred to as the internal versus external

view). The open world assumption makes the assumption that the data available will always represent a partial incomplete picture of the domain (e.g. multiple law enforcement datasets, focussing on a particular outlaw motorcycle gang, are to be combined), and the closed world assumption the opposite, that the domain is indeed contained within the data available (e.g. two retail businesses merge and want to combine their customer databases).

Interestingly, the graph also enables extending the notion of pair equivalence beyond a dyadic sense into the more contextual component-based view. This more contextual view, known as collective entity resolution, vastly improves classification / decision making and has been shown to improve efficiency and F-measure evaluations (Yongxin, Qingzhong, & Ji, 2009). Bhattacharya and Getoor (2006, 2007) intuitively applied collective entity resolution using hypergraphs; however a more contextual approach can be used simply by applying community detection, transitivity and logic to a standard property graph. Local versus global resolution is another related idea – where equivalence can be ring-fenced to a local portion of the graph in instances where it is appropriate (e.g. 'Andy SMITH' is only equivalent to 'Andrew SMITH' if proximal) (Bhattacharya & Getoor, 2006).

When viewing dyadic predictions as a graph, the value of graph-based metrics to identify error and measure accuracy of ER predictions becomes a natural extension (Naumann & Herschel, 2010; Randall et. al, 2014). Specifically, denser completely transitive components of predictions are more likely to be accurate whilst sparse components are more likely to contain error in the form of false positives and false negatives. Many metrics, such as diameter and transitivity, have been utilised to measure error successfully (Randall et. al, 2014).

Negative evidence is a critical element and is often applied early in the entity resolution process to constrain the number of pairs that are assessed for equivalence. This makes good sense from both a quality performance and computational performance perspective. The use of logic in constraints requires domain knowledge, an acute understanding of data quality, and the data representation derived from applying this knowledge to enable performance improvements. For example, in many instances using the gender attribute to support indexing is preferable, however if the quality of the gender attribute is "poor" then using the attribute as a constraint when indexing will introduce unacceptable error. Identity attributes can also be used further downstream in the process when classifying whether pairs are a match or not (or a potential match). An example of this is derived from the graph approach where the existence of a relationship, or particular set of relationship (e.g. "twin of"), between *i* and *j*, perhaps coupled with provenance attributes, can indicate that the pair cannot (or is highly unlikely to) be the same entity.

Of course any approach to determine equivalence between two references is dependent on the data they use as an input. More specifically an augmented data model, data quality, supplementary data

sources, and maximising the value elicited from domain knowledge - the building blocks of entity resolution. Linguistic research, and specifically onomastics – the study of proper names (a class of names that are uniquely identifiable) -, is one avenue to significantly augment this collection of data aspects. Having an explicit understanding of naming convention generally and the sources of variance generated from naming convention including, transcription, homophones, hypocorisms, and cultural variants, leads to a more comprehensive and accurate set of data inputs to perform entity resolution on (Lisbach & Meyer, 2013). For example, an enhanced understanding of name features such as structure (e.g. Arabic names constructed of Ism, Kunya, Nasab, Laqab, and Nisba), naming convention (e.g. the generational name will be shared by siblings), syllable count (e.g. Chinese names are generally monosyllabic), and hypocorisms (e.g. Mohd is equivalent to Mohammed) – which are alternate consistent nickname versions of a name - can lead to improvements in the data model, using additional sources of data to buttress performance (e.g. using a gazetteer to support address matching) and the generation of new metadata through the deployment of functions (e.g. identifying name origin) to support better performance. More generally relevant onomastic knowledge can not only augment the identity attribute stage, blocking, and/or algorithm configuration and parameters stage to improve performance, but can also guide the construction of specifically designed algorithms to target specific name variance (e.g. an algorithm that is tuned to identify pairs of entities where *i* uses a maiden name and *j* uses a married name), and be used as a suite of focussed algorithms.

Named entity recognition (NER) is a related area of research focused on the detection of proper names from unstructured text (Nadeau & Sekine, 2007). We mention it here because many ER models utilise overlapping approaches with NER – such as the use of gazetteers. Additionally, determining the origin of person proper names is a relevant research area for ER, which has been the subject of academic research from a NER perspective. The focus has largely focused on n-gram statistical methods (Nobesawa & Tahara, 2005) and an ME-based classification utilising an ontology including relationships between origin grammar and linguistic features, including n-gram (Fu, Xu & Uszkoreit, 2010). Both of these approaches showed encouraging results. The statistical n-gram approach of Nobesawa & Tahara (2005) attempted to identify the origin of proper names from 12 countries with accuracy ranging from 50 to 93%, and the ME-based classifier, utilising n-grams and linguistic features, attempted to identify the origin of proper names from 8 countries with accuracy ranging from 73 to 98%. Unfortunately, neither runtime nor scalability was mentioned in these papers, crucial elements to applied computational models.

In terms of entity resolution methods specifically designed and applied to the criminal domain Li and Wang (2013) proposed a graph-based approach using a hypergraph focussing on three types of attributes – personal identity, social behaviour, and social relationship. Personal identity refers to attributes such as name, date of birth and social security number; social behaviour refers to using

some behavioural element to assign group membership such as transactional or criminal pattern; social relationship refers to neighbourhood similarity. These three elements are represented as a hypergraph which is then used to determine co-reference via applying one of three matching strategies (pairwise comparison; transitive closure; collective clustering). Evaluation of Li and Wangs's (2013) approach on a small synthetic data set (*N = 100*) yielded a highest f-measure of 0.8939, using the collective clustering matching strategy. Unfortunately, real-world applicability, generalizability, runtime performance and scalability were untested.

In 2016 Robinson applied Reference Graphs (a knowledge graph), as a feature of entity resolution, to two real-world criminal datasets. The Reference Graph was generated via an algorithm that takes input data and gazetteer and generates a graph of proper names with edges representing similarity between those proper names. A pruning strategy and community detection were then applied to generate classes. Significant comparative benefits were found when using the Reference Graph Algorithm for blocking and decision management.

The Entity Resolution module of the GCND applies and extends many of these concepts including iterative resolution, collective entity resolution, neighbourhood similarity, negative evidence, linguistics and other graph concepts.

2.2.3 Link Discovery

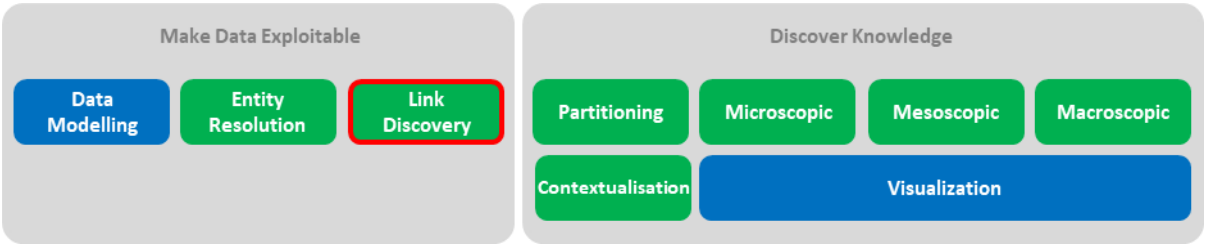


Figure 2.4. This figure outlines the modular design of GCND, with the current focus on Link Discovery.

Link Discovery is a subset of link and node discovery, which is further defined as the inference and prediction of unobserved real-world 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 nodes (i.e. nodes in the dataset but not in real-world) and spoof nodes (i.e. a real-world node is represented as one or more nodes in the data) (Maeno, 2009). Node discovery is largely undertaken as part of the entity resolution process (see above) so the focus here will be on link discovery.

Link Discovery in particular is a critical element to enhancing the quality and completeness of dark networks (Xu & Chen, 2008; Hu, Kaza & Chen, 2009). Often this step, if performed well, can transform data to a state where knowledge discovery can be meaningful. Independent of the knowledge discovery phase the derivation of hidden data can obviously provide significant value to any intelligence or investigatory activity at the most fundamental level (e.g. identifying that an entity is using an alias; identifying a link that indirectly connects a methamphetamine trafficker with a methamphetamine wholesaler). Let's now look at link inference (LI) and link prediction (LP) in turn.

Link inference

The inference of edges is defined here as using the data representation and logic to uncover implicit data that is not explicitly observed. This can be done through semantic logic leveraging off the data representation and through statistical inference, such as the Bayesian approach outlined by Rhodes and Jones (2009). An example of semantic logic could be an intransitive relationship where it is identified that *i* is the child of *k* and *j* is the child of *k*, but we do not explicitly have an edge between *i* and *j*. In this case it is simple to deduce that the transitive relationship between *i* and *j* is that of sibling. Another area of research in link inference is based on co-occurrence inference utilising bipartite graphs (also known as affiliation networks), which requires optimisation when edge weight and type become important considerations (Latapy, Magnien, & Del Vecchio, 2008). In later chapters we will outline how using inference can make implicit relationships explicit using directed graphs and bipartite graphs. Another body of research, known as "concept space", involves the inference of relationships between entities by their co-occurrence in relevant text. This approach can be as simple as counting the times that a pair of entities co-occurred in a corpus of documents or may be more sophisticated in terms of how the text set is defined – i.e. by sentence or paragraph (Xu & Chen, 2003).

Link prediction

The prediction of edges - link prediction - is often couched in the context of dynamic graphs and the prediction of future links, however this notion can be extended to apply to static and dynamic graphs in which the open world assumption presides and there is significant data incompleteness (Liben-Nowell &Kleinberg, 2007; Rhodes & Jones, 2009; Guimera & Sales_Pardo, 2009). So, link prediction is applied here, not as predicting future relationships but predicting what relationships exist in the real-world but are merely unobserved in the data.

People typically form voluntary relationships through two closure mechanisms; cyclic closure which is a relaxation of triadic closure, where transitive relations occur (*i* is related to *k* and *k* is related to *j*, then *i* is also related to *j*), and focal closure, premised on homophily, where two entities form

a relationship through a common focus (e.g. sport, religion, profession) (Holland & Leinhardt, 1971; Feld, 1981; Kossinets & Watts, 2006). People use their neighbours and communities as reference points from which to form new connections. These contextual reference points drive cyclic closure. Focal closure however is driven predominantly through homophily and is categorically based assessing "fit" via a series of elements (e.g. age, gender, socioeconomic status, nationality, ethnicity, religion, occupation, education, geographical proximity, criminal experience, criminal status) against the goals of the entity - friendship, getting access to specialist knowledge, or status for example (Giuffre, 2013).

We can use this understanding of relationship generation mechanisms and identify manifestations of these mechanisms as topological features; approximately scale-free degree distribution and assortativity. An approximately scale-free degree distribution is found in many social graph topologies, with preferential attachment identified as a key driver (Barabási & Albert, 1999). Preferential attachment is the concept that those entities that substantively have more of something will in future generate more of that something in comparison to entities that substantively have less of that thing, and is also known as the "cumulative effect" and the "rich get richer effect" (Barabási & Albert, 1999). So, high degree nodes will tend to have more unobserved edges than low degree nodes. Assortativity, or assortative mixing, is the observation that entities that are connected tend to be similar, which ties into the premise that homophily drives the generation of relationships (McPherson, Smith-Lovin & Cook, 2001; Newman, 2003b).

Published applied link prediction computational methods typically consist of generalisable heuristic similarity metrics utilising local and global topology and more domain specific statistical / machine learning model utilising a range of features. The features are often a conglomeration of similarity metrics and properties selected in the context of known assortative elements, such as age. Studies by Holland and Leinhardt (1971) and Davis (1979) also illustrated that the structure of the network can be inferred by the triadic structure present in a sample network, however the application to real-world data is untested in this setting.

Topology metrics are commonly applied to graph problems, and can be grouped into assortativity methods, neighbourhood similarity methods, path-based methods, partitioning and hybrid methods.

Assortativity measures typically look at attribute similarity, in the context of whether that attribute is expected to be assortative within the network, or proven to be assortative within a training set. For example, people are found to display a marked preference for having connections with people of a similar age, as was vividly found in the analysis of Facebook users (Ugander, Karrer, Backstrom, & Marlow, 2011).

Neighbourhood similarity methods, as explained above in more depth, include common neighbour, Jaccard coefficient, and Adamic-Adar metric. An interesting extension from the field of genomics and systems biology is the Topological Overlap Measure (Ravasz, Somera, Mongru, Oltvai, & Barabási, 2002), and its generalization, the Generalized Topological Overlap Measure (Yip & Hovarth, 2007). This set of metrics are generally deployed in a very local sense to an order of 2.

Path-based methods include shortest path or graph distance, Katz (Katz, 1953), PageRank (Page, Brin, Motwani & Winograd, 1999), and SimRank (Jeh & Widom, 2002). Path-based metrics suffer from the small-world nature of many graphs in that the meaningfulness of paths seems to quickly decay (Liben-Nowell & Kleinberg, 2007), rendering these metrics susceptible to error unless configured appropriately.

Partitioning is another method of generating context of the contextual proximity of the local graph surrounding a pair of nodes, whether that approach is using community detection, clustering (Li, He, Huang, Zhang & Shi, 2014), or stochastic blockmodeling (Guimerà & Sales-Pardo, 2009).

Maximum likelihood estimation is a statistical method used to generate a likelihood for non-existent edges (Clauset, Moore & Newman, 2008). Hybrid methods include an extension or combination of the neighbourhood similarity and path-based methods, with assortative approaches.

We will now cover in more detail some of the key metrics that are of specific relevance to the GCND.

The common neighbours method simply counts the number of shared neighbours between a pair of entities, or in other words counts the length of the set of entities that are derived from the intersection of the neighbours of the vertices *u* and *v*. Newman (2001a) found a correlation between common neighbour measures and future probability of co-authorship.

$$score(u, v) = |N(u) \cap N(v)| \qquad \text{Eq. (1)}$$

The Jaccard coefficient is an extension on the common neighbours approach, taking into consideration the number of connections the pair of vertices have. This makes intuitive sense particularly in the context of the scale-free topology of social networks. The measurement is the intersection of the neighbours of the vertices *u* and *v* (i.e. their common neighbours) divided by the union of all neighbours of vertices *u* and *v*.

$$score(u, v) = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|} \qquad \text{Eq. (2)}$$

The Adamic-Adar method incorporates the concept that a pair of vertices are more similar if the neighbour(s) they share have relatively few connections as those highly connected vertices are probabilistically more likely to be a shared neighbour (Adamic & Adar, 2003). The metric refines the Jaccard coefficient by weighting less connected vertices more heavily.

$$score(u,v) = \sum_{z \in N(u) \cap N(v)} \frac{1}{\log(|N(z)|)} \quad \text{Eq. (3)}$$

The Topological Overlap Measure (TOM) is a variation on this theme by taking into account the presence or absence of a link between the pair of vertices and additionally normalises the metric (Ravasz et al., 2002).

$$score(i,j) = \sum_{u} a_{iu}a_{uj} + a_{ij} / \min(k_i, k_j) + 1 - a_{ij} \quad \text{Eq. (4)}$$

The generalization of the Topological Overlap Measure, the Generalized Topological Overlap Measure (GTOM), enables the neighbourhood concept to be extended by propagating through the neighbours to a fixed order (e.g. an order of 2 would include the neighbours of neighbours). The GTOM is defined as follows (where *x* represents the order of neighbourhood):

$$GTOMx(i,j) = \frac{|N_x(i) \cap N_x(j)| + a_{ij}}{\min(|N_x(i)||N_x(j)|) + 1 - a_{ij}} \quad \text{Eq. (5)}$$

Path-based methods include the Katz, rooted PageRank, and SimRank metrics. The Katz metric is based on measuring all walks between a pair dampening the value of long paths so the shortest paths contribute exponentially more (via an attenuating factor β).

$$score(x,y) := \sum_{\ell=1}^{\infty} \beta^{\ell} . \left| paths_{x,y}^{(\ell)} \right| \quad \text{Eq. (6)}$$

PageRank can be adapted to the link prediction problem – rooted PageRank (Liben-Nowell & Kleinberg, 2007), and SimRank is a recursive based approach that is a generalization of common neighbours that walks between a pair of nodes iteratively assessing similarity and then using this new knowledge to assess similarity in a recursive manner (Jeh & Widom, 2002).

Partitioning approaches are an interesting area of research. From a proximity perspective community detection (non-overlapping and overlapping), clustering and stochastic blockmodeling are three methods that can generate additional local graph context to make a better decision on whether it is likely an edge exists between a pair, but also methods that detect structural position in a stochastic

sense can generate significant mesoscopic based context to contribute to predicting links (White, Boorman & Breiger, 1976; Guimerà & Sales-Pardo, 2009; Guiffre, 2013). Indeed, community detection can be used to not only measure proximity between a pair of entities but also from a mesoscopic perspective can detect proximity between communities (Giuffre, 2013). There is a clear application of overlapping community detection and non-overlapping hierarchical community detection algorithms in this space.

The metrics mentioned above can all be used to predict unobserved edges between nodes. The framework most commonly utilised to optimise the prediction of unobserved edges is machine learning. This consists of taking a set of data features and engineered features (e.g. similarity metrics), identifying a representative training set, optimally tuning the machine learning model, deploying the optimised model (or models if ensemble or meta-models used) and use the pairwise probabilities generated to predict unobserved real-world relationships, perhaps by using a threshold. Ensemble or meta approaches that make use of multiple variables, such as those generated through the methods discussed earlier are gaining favour (Hasan, Chaoji, Salem & Zaki, 2006). Similar to entity resolution approaches is the clear fact that the thresholds used in determining whether a pair of nodes are equivalent or whether an edge does exist between a pair is dependent on the context and domain of the implementation. Some clients will require a conservative threshold so as not to introduce too much error (e.g. generating targets for investigation of money laundering) and some will require an aggressive approach to counter a lack of fundamental data (e.g. counter-terrorism). Furthermore, the quality (e.g. confirmed, unconfirmed, vague or tentative), weights, source, or semantic edge type (e.g. phone call, familial, cohabitation, co-shareholder) should be taken into consideration, when making decisions on predicting links, and if possible enhanced as much as possible. Later chapters will explicitly cover the link discovery approach implemented in GCND.

In terms of link prediction applied to a criminal setting four studies stand out. Hu, Kaza and Chen (2009) used statistical procedures (Multivariate Cox Regression) on a dark network to empirically determine the most useful variables for predicting links and found that the presence of mutual acquaintances and "sharing a vehicle" were statistically significant features. Rhodes and Jones (2009) used a Bayesian inference model to detect unobserved edges in a terrorist group (22 persons) and generated accuracy rates of between 0.26 and 0.45 in the assessment of 136 pairs. Fire, Puzis and Elovici (2013) built a machine learning model using topological features using an open source terrorist dataset (244 vertices and 840 edges). However, Fire et al. (2013) used AUC (area under the curve) as the sole metric, which while convenient, fails in this context to mimic how the technology would be deployed in the real-world. Berlusconi, Calderoni, Parolini, Verani and Piccardi (2016) built a topologically based approach using common neighbours, Katz index similarity and Structural Perturbation Method (Lu, Pan, Zhou, Zhang & Stanley, 2015) predicting marginal links actually

removed throughout the investigation / judicial process on an organised crime network (182 vertices and 549 edges). Berlusconi, and co-authors, (2016) focus was also not so much link prediction but ranking edges previously discarded for their inferred importance.

The results from these studies all give some comparative basis from which to judge LP performance, however these studies fail to demonstrate applied scalability and generalisability. Two critical components to the performance of a scalable applied LP model.

The core modules of entity resolution and link discovery, in the context of creating an appropriately explicit and efficient data representation and integrating relevant cleaned datasets creates a data asset that is maximally exploitable and creates the opportunity to build on the explicit data to discover latent knowledge.

2.3 Discover Knowledge

Knowledge discovery can be defined as the application of methods to extract latent knowledge from a body of data. Often knowledge discovery is coupled to data mining however we will avoid the term due to the ambiguity attached to it. Knowledge discovery in this context is interpreted and applied as using a computational approach that exploits the data maximally to generate novel knowledge given the trade-off decisions around computational expense, computational complexity, ease of explanation, ease of conveying findings, generalizability of findings, reusability, maintenance, etc. All of these factors are context dependent so we have made judgement calls throughout in terms of balancing these elements. We conceptualise GCND as following a life-cycle (see Figure 2.5., Figure 2.8., Figure 2.9. and Figure 2.12.) and as such will talk about downstream and upstream aspects. Often improvements upstream in improving the quality of the data and the way it is represented can create more value than extending a knowledge discovery model. And of course making a positive change upstream can potentially serve to enhance the output of multiple models. We use the term model, function and metric interchangeably as we conceptually view any method type to extract knowledge as fundamentally analogous. A core belief that influences the choice of functions that have been deployed is that modelling is not about the data but about the construct you are attempting to measure and in alliance with data you can create a computational solution to, in part, explain the construct. This is an iterative and exploratory endeavour that can be extended through enhancing data, the way it is represented and the methods used to gain insight. Therefore, methods such as neural networks need to be used in a way that enables interpretation of the mechanisms of the construct. We have broken knowledge discovery in to four parts; partitioning graphs, contextualisation, microscopic knowledge discovery, and mesoscopic and macroscopic knowledge discovery. These aspects are however deeply intertwined. Within any complex systems approach these elements will be effectively omnipresent.

The core underlying premise is that the problem needs to be always placed in context. Context in terms of the domain context and the context that the problem is viewed in situ - part of an interconnected system.

2.3.1 Partitioning graphs

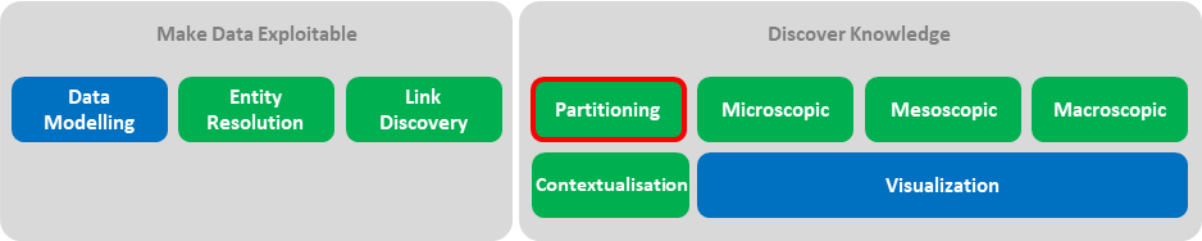


Figure 2.5. This figure outlines the modular design of GCND, with the current focus on Partitioning.

The partitioning of graphs is an area extensively researched, as it reflects a core aspect to uncovering contextual knowledge about problems that can be represented as graphs. Components are perhaps the simplest graph partition. Components are a subset of nodes from a graph that form a connected subgraph that is unconnected to the rest of the graph. The giant component refers to the largest of these unconnected subgraphs. Community detection is the set of methods used to determine node membership based on the relationships between nodes (Wasserman & Faust, 1994). The definition of community detection is however not formalised but merely reflects the intent to group nodes into communities (or modules) that exhibit more intra-community relationships than inter-community relationships (Newman, 2004). There are diverse applications of community detection across a range of domains including social, information and biological networks. The partitioning of graphs can also be of interest from perspectives other than the clustering of inter-related nodes. The second area of the partitioning of graphs under focus is the identification of classes of nodes based on the equivalence of the pattern of relationships they exhibit – equivalence classes. The concept of position encompasses this notion, and is underpinned by a range of methods developed to measure this equivalence, including structural equivalence (Lorrain & White, 1971), regular equivalence (White & Reitz, 1983; Borgatti & Everett, 1992a), and automorphic equivalence (Winship, 1988; Mandel, 1983; Winship & Mandel, 1983; Borgatti & Everett, 1992a). The extension of position is the concept of role which goes beyond relation equivalence to include the context of a position to identify a set of nodes sharing a functionally similar role (Lorrain & White, 1971). The third area of partitioning graphs can be referred to as graph mining – the detection of patterns in graphs. This can include blockmodeling (see below) but often encompasses frequent subgraph mining (FSM) and graph clustering.

Graph partitioning is used extensively throughout GCND across every module as the notion of groups is fundamental to a complex systems perspective. The Entity Resolution module specifically utilises partitioning frequently to maximise notions of graph distance and communities to make better entity resolution decisions.

Community detection

As expressed earlier, the goal of community detection is to ascribe membership to nodes based on the pattern of relationships they have in relation to each other. With those nodes that have a dense set of edges between them (intra-community edges) classified as a community relative to those same nodes lack of connections to other clusters (inter-community edges) of nodes that have a dense set of edges between themselves. This pattern of communities (or modules) has been demonstrated across a range of networks including social (citation, mobile phone use) and biological (marine organisms, protein complexes, neurological) networks (Newman, 2003a; Blondel, Guillaume, Lambiotte, & Lefebvre, 2008; Ravasz et al., 2002; Krogan, Cagney, Yu, Zhong, Guo, Ignatchenko, & Greenblatt, 2006).

Modularity is a key measure of community that measures how strong the intra-community set of edges are relative to the sparsity of inter-community edges. It remains a key metric of measuring and evaluating community detection, however the use of modularity as a metric to determine fine-grain community structures has been shown to be deficient (Fortunato & Barthelemy, 2007). Fundamentally community detection algorithms can generate non-overlapping or overlapping memberships. Each approach has valid applications with a series of strengths and weaknesses. Non-overlapping approaches are fundamentally simpler to compute, less expensive, and simpler to represent, however in many instances may be an over-simplification of the real-world phenomenon under consideration. This oversimplification is based on the simple assertion that each of us is a member, whether we consciously acknowledge or not, in multiple communities, such as immediate family, extended family, co-workers, sports team, suburb, and neighbourhood progressive dinner club. These communities infer differing types and strengths of attachment, influence, dynamism, and goal, however nonetheless the point is clear that these phenomena exist and impact on micro behaviour and access to resources etc. Therefore, overlapping and nested communities are a more natural representation of the real-world, but introduce more complexity. Fortunato (2010) provides an excellent exposition of community detection and the methods used to detect communities. We will now cover a small fraction of the methods that have been developed.

Non-overlapping community detection algorithms

The Fast Greedy (Newman, 2004; Clauset, Newman & Moore, 2004) algorithm optimises the modularity score (Q) using an approximate optimization greedy algorithm, comparing the fraction of

edges between group of nodes *i* and group of nodes *j* with what would be randomly expected with a community that had no more edges than by random chance receiving a *Q* score of 0 (Newman & Girvan, 2003). In practice a *Q* score over 0.3 indicates a significant community structure. Computationally this algorithm is relatively inexpensive; however it has the restriction of not working on directed graphs.

The InfoMap (Rosvall & Bergstrom, 2008) algorithm detects communities through finding an optimal compression of the graph topology. This is achieved through undertaking a random walk through the graph and efficiently describing this walk (using Huffman code) allotting unique codes to every node and efficiently describing the entry and exit point of modules, which then enables efficient coarse graining of the network into modules. Computationally this algorithm is relatively inexpensive; and it has broad applicability on weighted and directed graphs.

The Louvain (Blondel et al., 2008) algorithm uses modularity in an agglomerative hierarchical approach. Every node is assigned to unique communities and then agglomeratively re-assigned to a new community which maximises its contribution to modularity. When node reassignment is complete the community is contracted and the process is restarted using the merged communities. Computationally this algorithm is similarly inexpensive as the Infomap and Fast Greedy algorithms (above); and it has broad applicability.

Overlapping community detection algorithms

The Clique Percolation (Palla, Derényi, Farkas & Vicsek, 2005) method detects communities by identifying a union of adjacent k-cliques, where those k-cliques are assessed as being adjacent if they share $k-1$ nodes. In other words identified k-cliques roll over adjacent k-cliques and are aggregated into a community if they share *k-1* nodes. Computationally this algorithm is relatively expensive to the point that it is not scalable, and given the size of network within this domain is not viable.

The LinkComm (Ahn, Bagrow & Lehmann, 2010) algorithm focuses on identifying community structure in edges rather than nodes. The method employs the Jaccard coefficient for assessing the similarity between edges that share a node, assigns a pairwise similarity to all network edges and then hierarchically clusters and cuts the dendrogram at maximal density. This enables the identification of a node's presence in multiple communities, as nodes would be expected to have multiple edges, identifying over-lapping and nested structures. Computationally this algorithm is relatively inexpensive when compared to other non-overlapping methods, however it is still expensive compared to non-overlapping methods. The computational expense does not rule it out but obviously limits its utility somewhat.

Applying community detection to criminal networks

Jenkins and Potter (1987) amongst the majority of commentators (e.g. Natarajan, 2000; Morrison, 2002; Morselli, 2009; OFCANZ, 2010; Calderoni, 2011; Abadinsky, 2012; Savona, 2012; Siegel, 2012; Galeotti, 2012; Lo & Kwok, 2012; Albanese, 2012; Europol, 2013) articulate and clearly demonstrate that organised crime is predominantly formed by loosely organised fluid functional groups of entities rather than bureaucratically defined hierarchical structures. These fluid groups are responsive to opportunities and engage in overlapping endeavours (Paoli, 2002).

Indeed, there is a body of empirical evidence that organised criminal groups, such as outlaw motorcycle gangs, do not functionally organise around the structure of the gang, but each entity within the gang utilises his own extended neighbourhood to create functional groups to engage in activity, whether illegal or not (Tusikov, 2010; Siegel, 2012). In fact Morselli (2009) found that the gang structure was not a coherent social entity and that subgroups of the gang are just as important to the overall gang, and interestingly gang activity was driven through brokerage and cut points.

Applying community detection in this context poses problems of accurately determining the boundaries of communities, identifying non-obvious members of "functional groups", and doing so within the context of potential conflation where group size is over stated (Bichler & Malm, 2015). So it is very important to clearly state that the goal of community detection is not to in isolation identify gang membership or functional group membership but gives us network structural membership to enable context to measure and understand the domain in greater detail.

Equivalence classes

Equivalence refers to similarity, and in this case similarity of nodes with the goal of grouping nodes into classes based on those that are equivalent. In the context of graphs equivalence can be conceptualised as based purely on network features – position – or in the context of the domain and edge and node attributes – role. Position refers to a network position that is occupied by a class of nodes that are inherently similar. This similarity can be measured using a group of equivalence metrics that have varying levels of strictness – structural equivalence, regular equivalence, automorphic equivalence, and isomorphic equivalence.

Structural equivalence is the notion that two nodes are structurally equivalent if they share exactly the same relationships to and from identical nodes. Nodes are not substitutable. This restriction to identical edges and nodes restricts the notion locally as for example two teachers can only be structurally equivalent if they teach the same students. This means the notion of structural

equivalence does not generalise outside these local areas across the graph, and does not allow comparison between graphs (Faust, 1988; Borgatti & Everett, 1992a; Wasserman & Faust, 1994).

Isomorphic equivalence is when two sets of nodes in different graphs have the exact same pattern of relationships, preserving the property of adjacency, ignoring atomic level features of nodes and edges. Automorphic equivalence is analogous to isomorphic equivalence however the mapping is not from one graph to another but is from a graph back onto itself. Having the exact same pattern of relationships means that for two nodes to be automorphically equivalent they need to have the exact same set of graph theoretic properties, including in-degree, out-degree, betweenness, etc. (Borgatti & Everett, 1992a). The only aspect that can be different between the two nodes, and remain automorphically equivalent is that graph labelling is substitutable. Automorphic and isomorphic equivalence is notoriously expensive to measure, and although heuristic algorithms have been developed to increase performance they too are still computationally expensive (Wasserman & Faust, 1994). Again these notions also suffer in terms of their applicability to the real-world, specifically when considering incomplete graphs with significant error.

Regular equivalence does not have the restriction of having identical edges to other nodes (structural equivalence), nor have the exact structural pattern (as is required by automorphic and isomorphic equivalence), but focuses on having identical relationships to and from other actors who share the same equivalence class (White & Reitz, 1983). So, for example (as outlined in Figure 2.6.), two teachers that teach a different set of students are regularly equivalent as the students share the same equivalence class. Importantly, the number of relations between equivalence classes is not considered when measuring regular equivalence. From a neighbourhood perspective regularly equivalent actors need to have their neighbours come from the same set of equivalence classes.

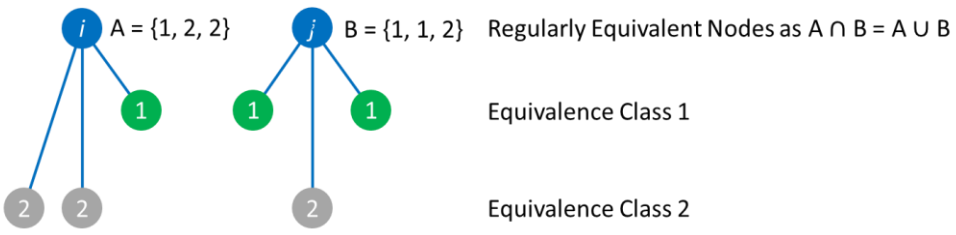


Figure 2.6. Neighbourhood perspective of an example of regular equivalence.

Blockmodeling

Blockmodeling (including generalized blockmodeling) is an empirical modelling approach that permutes the order of the rows and columns of a matrix to uncover some knowledge about the structure of the matrix (or graph), and reduces this matrix into a form that retains the structural

features and enables interpretation. The grouping and reduction is based on the extent that pairs are equivalent, whether that equivalence is based on community or some other notion such as regular equivalence. The reduction, or contraction, of the nodes into classes then enables inter and intra class relationships to become obvious.

The first step of blockmodeling is to attribute an equivalence class to each node based on some notion of equivalence, usually through the application of an equivalence algorithm (e.g. REGE) which generates a dissimilarity matrix, which requires a clustering approach and potentially a tree cutting approach and some domain knowledge to determine an optimal set of classes. Within generalized blockmodeling this step can be calculated directly with a partition generated with an evaluation of the partition via (f-)regular equivalence. The second step is determining the extent of inter and intra class relationships. This is done by permuting the adjacency matrix so the adjacency of rows and columns are based on nodes of the same class, and then using this to generate a density table, image matrix and/or reduced graph to enable interpretation.

Each of these representations contracts the nodes into equivalence classes and represents the inter-positional and intra-positional relationships. A density matrix is constructed from calculating the density of each submatrix, ignoring diagonal (loops) cells. So a density matrix will represent each equivalence class (or position) in the form of a matrix and give the density for each inter-positional and intra-positional relationship. This can then be parsimoniously transformed into an image matrix of either 0's and/or 1's or block descriptions (e.g. complete or null). However, in reality each submatrix is likely to be somewhere between a complete (i.e. density of 1) and null block (i.e. density of 0). So, criteria is required to assign a 1 or a zero to a block that is somewhere in between. Many variants include; regular blocks that have a 1 in at least every row and column, row-regular blocks that have a 1 in at least every row, column regular blocks that have a 1 in at least every column, and density criterion, where a simple threshold is used to assign a block a zero or a 1. A reduced graph is merely a graph representation of the image matrix.

Stochastic blockmodeling (SBM), a type of random graph model, is an alternative to the empirical modelling of blockmodeling. SBM provides a direct approach to determining equivalence classes setting an upper and lower bound (Q) to guide the number of partitions, and secondly generates estimates of the posterior probability of class membership enabling class attribution, and thirdly enables the generation of entropy and other potential evaluative scores to measure the accuracy of the model.

The interpretation of blockmodels is derived from hypothesising from the perspective of position or role. From a position perspective each class can be understood and labelled from their inter-class and intra-class local and global positional context. For example, figure 2.7. illustrates how the

interpretation of the function of each position is based on in-degree and out-degree, and is dependent on both local and global network context.

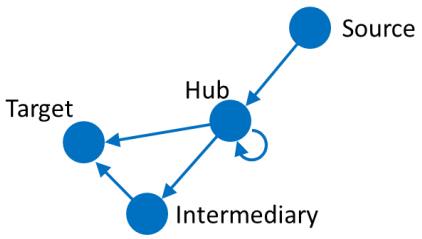


Figure 2.7. Reduced graph of blockmodel.

The image matrices can also be used to hypothesise from a global perspective. Faust and Wasserman (1992) detail numerous patterns that are designed to explain the structure of the classes in relation to each other, with labels that include; cohesive subgroups, center-periphery, centralized, hierarchy and transitivity. From a role perspective the method of interpretation is based on looking for a relationship between node attributes and classes to give a contextual understanding of the potential latent roles that have been uncovered. To support this endeavour pre-processing of data can deliver rich rewards. We refer to this as contextualisation – the generation of explicit data to place context on the interpretation of modelling.

Graph mining

Graph mining is broadly defined as the detection of patterns in graphs, however here we take a narrower view so graph mining is defined as the identification of relevant subgraphs within a graph. The detection of relevant subgraphs, or more simply put the detection of groups of persons coalescing for criminal purposes, is of pronounced importance.

Graph mining and unsupervised learning approaches in general are uncommon within law enforcement and intelligence agencies. Often data is fundamentally used as a reference set from which query-based approaches can inform risk management decisions (e.g. intelligence collection or investigation targeting). The query paradigm can evolve into using data to inform the development of rules-based approaches, and more advanced capabilities utilise supervised learning approaches to target specific risk-based problems (e.g. missing trader fraud). Graph mining can be deployed using a raft of methods, however the most common are frequent subgraph mining and clustering, so let's focus on these.

Frequent subgraph mining (FSM) primarily bases pattern detection on the topological structure of a subgraph. This may be in the form of inferring specific topological patterns (Prado, Plantevit, Robardet & Boulicaut, 2013) or the identification of subgraph isomorphisms (Junttila & Kaski, 2007).

Clustering is focused on generating vertex classes, or in other words placing those vertices that are deemed similar into specific classes. The specific relevant task here relates to generating overlapping classes (Palla et al., 2005) and identifying those singleton vertices. Structural and regular equivalence (usually deployed by blockmodeling or stochastic blockmodeling) form another group of relevant approaches. Equivalence based approaches use both topological structure and clustering to assign classes, and can conceptually detect graph structure and specifically classes of role (Wasserman & Faust, 1994).

The scale, incompleteness and uncertainty typically encountered within the criminal domain renders the application of FSM and clustering approaches problematic. Specifically, these methods rely on a well-defined limited context allied to highly curated data, including explicit semantic edge labels, to enable accuracy. If these elements are not present these methods will simply not perform. Xu and Chen (2005) successfully executed a range of SNA metrics, including blockmodeling, on two criminal graphs that contained 57 vertices and 60 vertices. In this work, whilst demonstrating the potential of blockmodeling within the criminal domain on small subgraphs, the computational limitations were clearly demonstrated.

Let's now review the graph-based computational approaches that have been applied to the criminal domain.

A collaborative effort between Arizona State University, Tuscon Police Department, and Phoenix Police Department, starting in 1997, developed the COPLINK software. This research group has since contributed a significant amount of literature on utilising SNA and associated methods in the criminal domain and enhanced our understanding of crime (Chen et al., 2004). The focus of COPLINK did not include the detection of criminal subgraphs but rather focussed on the execution of a range of topological metrics on subgraphs subsequent to their identification (i.e. knowledge discovery) (Xu & Chen, 2005; Xu & Chen, 2003). Specific areas researched included link analysis (Chen et al., 2004; Schroeder, Xu, Chen & Chau, 2007), topology (Xu & Chen, 2008), and the identification of significant facilitators in evolving criminal networks (Hu et al., 2009). Typical of the size of subgraph analysed was that by Xu and Chen (2005) who examined two graphs of 57 vertices and 60 vertices.

GANG (Shakarian et al., 2015) is another software product that generates a set of SNA metrics on a predefined criminal group. GANG takes a well-defined criminal group as an input and partitions this subgraph using the Louvain community detection algorithm (Blondel et al., 2008) to create an 'ecosystem' view of how smaller communities within the group interact, and provides a set of SNA metrics. Shakarian, and co-authors, (2015) tested GANG on a 1,468 vertex graph. Both COPLINK and GANG are not designed to detect criminal subgraphs from large fused data. Rather they are

designed to take a small criminal group as an input and then generate a set of useful SNA metrics on that criminal graph.

Supervised learning graph mining approaches have been devised to detect terrorist groups via role-based approaches (Shaikh, Wang, Yang & Song, 2007), detect fraud using graph-based anomaly detection (GBAD) (Mookiah, Eberle & Holder, 2014; Huang, Mu, Yang & Cai, 2018), and use graph isomorphism to detect suspicious transactions (Michalak & Korczak, 2011). These approaches rely on specific, small-scale, high certainty and complete data which precludes these techniques from having utility in the scope of this work.

Two groups of researchers have created supervised learning approaches that utilise a 'database of crimes' that includes offence, co-offending, location, temporal, modus operandi, and offender name similarity to cluster the known criminal events. The 'combined detection model for criminal network detection' (ComDM) was developed by Ozgul, Erdem, Bowerman and Bondy (2010) as a conglomeration of previous models (GDM, OGDM, and SoDM) developed by the same research group. Wang, Rudin, Wagner and Sevieri (2015) created a space clustering method focused on the identification of crime series perpetrated by the same entity of group.

Li, Cao, Qiu, Zhao and Zheng (2017) developed an unsupervised approach to identify groups involved in money laundering using a temporal-directed version of the Louvain algorithm (Blondel et al., 2008), deployed on a distributed computing platform (Apache Spark). The approach takes a set of transactions and generates a multi-edge graph, transforming into a simple weighted graph, filtering any edges that represent a single transaction. Maximally connected subgraphs are extracted and partitioned using the temporal-directed version of the Louvain algorithm designed for the money-laundering domain. Communities are then assessed for number of nodes, number of edges, sum of money, average node degree, and temporal entropy enabling a ranking based on weighted rules. This method, whilst scalable and performant on the example provided, is tightly coupled to the domain. Additionally, the two underpinning assumptions that less complex communities are more likely money laundering gangs and hub nodes indicate a higher likelihood of money laundering are crude, unproven and misleading ensuring this method is not generalizable. A second significant issue is the performance degradation of community detection algorithms on dense graphs (Fortunato, 2010). Due to the dependence on community detection this approach is therefore exposed to volatile performance on graphs of variable topologies.

The rest of the literature that focuses on graphs (or networks) and the criminal domain focuses on testing specific hypotheses in relation to how a range of SNA metrics can be interpreted when applied on small criminal subgraphs. The criminal subgraphs studied almost always are highly curated datasets that are analysed in isolation out of their natural context. In other words the boundaries of the

criminal groups are simplified for the purposes of analysis. And in each case there is absolutely no attempt at detection of criminal groups. For example, Carley, Reminga and Kamneva (1998) focused on destabilising terror networks with a scalability of ~1,000 vertex graph. In 2002 Krebs (2002) undertook SNA analysis on the 9/11 network (37 vertices), in 2010 Morselli carried out SNA metrics on criminal groups from 25 vertices through to 174 vertices. Everton (2013) demonstrated the value of SNA in a case study on the Noordin Top terrorist group (79 vertices), and Morselli, Grund and Boivin (2015) analysed network stability on a co-offending network of 113,000 vertices. The theme running through this body of research is applying a range of SNA metrics to small well defined discrete highly curated criminal subgraphs.

2.3.2 Contextualisation

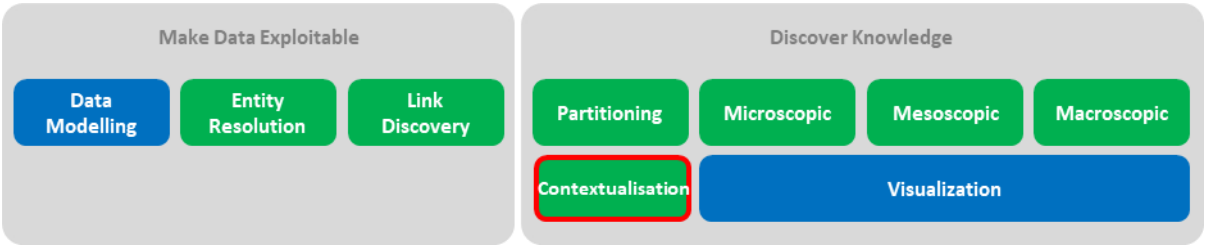


Figure 2.8. This figure outlines the modular design of GCND, with the current focus on Contextualisation.

The contextualisation of the problem is a crucial step to optimise the discovery of meaningful knowledge. Fundamentally, opportunities need to be explored, balancing three aspects; assessing data that is available, develop an acute understanding of what the key contextual concepts are, and an awareness of what technologies are available to construct new metadata from these two elements. The output of this contextualisation is larger in concept than just as an input for a model. It can serve to direct a whole branch of modelling opportunities and places constraints and context on the problem space to enable modelling to be coupled to the problem and meaningful to the consumers. In doing this there should also be no barrier to generalize the approach, to some degree, if modelled thoughtfully.

Supply Chain and Commodity flow

The modern marketplace is an integrated system that involves transactions between dependent firms. A supply chain is a specific component of this marketplace that involves a set of synchronised inter-related processes involving the acquisition of raw materials, the processing of those raw materials into a product, the distribution of these products, and crucially the facilitation of information exchange between dependent phases (Lambert & Cooper, 2000). Understanding the problem within the purview of the supply chain creates the direct context to frame the problem as a complex system and

immediately provides context to structural graph features. Furthermore, as the criminal community engages in more complex activity the more the system becomes interdependent because each entity is reliant on other specialised skill, knowledge or resource. Desroches (2005) found topological differences in each supply chain phase, with trafficking and wholesaling phases found to be more connected, with higher redundancy, higher status, smaller denser cliques, and more stable. The following features of groups involved in trafficking and wholesaling phases have been determined from empirical studies; drug markets are controlled by small groups and connected entrepreneurs (Natarajan & Belanger, 1998; Benson & Decker, 2010), opportunities are derived from familial or ethnic relationships (Morselli, 2005), a flexible division of labour where roles are interchangeable (Gimenez-Salinas Framis, 2014), and wholesalers play a significant role between traffickers and retailers (Adler, 1985; Gimenez-Salinas Framis, 2014).

From a profit perspective traffickers and wholesalers generate the highest profit per person, with retailer's profits somewhat diluted (Boivin, 2013). However, there is evidence that transnational trafficking groups are structured to funnel the majority of the profits from the active participants back to centralised entities, perhaps for capital investment in illicit or licit sectors, or for conspicuous consumption.

Based on research on Italian organised crime, Savona (2012) interestingly states that groups that maintain vertical integration or control whole aspects of the supply chain (e.g. by geography or commodity) generate a higher probability of engaging in corrupt relationships.

A number of studies have adopted a supply chain model and applied it to an illicit market within a broader network model and found varying topological patterns and network resilience (Malm & Bichler, 2011).

Incompleteness

The incompleteness of data can be measured in a passive and active way. Passively various expert-based information sets or topology based metrics can be deployed to assess the completeness and quality of data. In an active sense however sampling and simulating the impact of missing data can be used to measure the robustness of data and the reliability of functions across varying levels of incompleteness (Carley, Lee, & Krackhardt, 2002). The identification of structural holes has been another strategy proposed to identify data incompleteness and guide further data collection activity (Xu & Chen, 2005).

2.3.3 Microscopic Knowledge Discovery

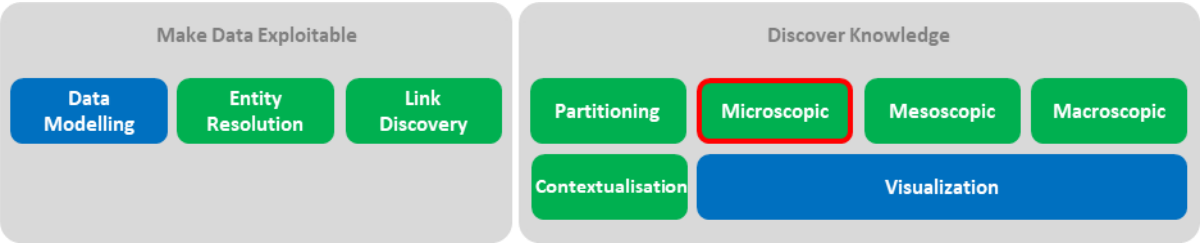


Figure 2.9. This figure outlines the modular design of GCND, with the current focus on microscopic knowledge discovery.

Microscopic Knowledge Discovery refers to the discovery of knowledge at the level of the entity, relative to other entities. Importantly, this knowledge still needs to be understood in the context of the domain and mesoscopic and macroscopic levels. We will now briefly survey some relevant metrics that are used within GCND at both the Discover Knowledge phase and specifically within the Entity Resolution module. The application of these metrics within the criminal domain literature will be specifically noted.

Centrality

Centrality finds its roots in SNA and in its broadest sense is about identifying the most important or prominent nodes. Vast arrays of metrics have been developed from a network perspective to measure, or more accurately, provide a relative rank of prominence. The interpretation of prominence is context specific and this class of metrics enable the interpreter a graph centric perspective. Perhaps more powerful is when they are analysed in combination, with attribute and contextual information, to advance understanding of the topic domain.

Importance can be viewed as either the transfer or flow across the graph (Borgatti, 2005) or a node's contribution to the cohesiveness of the network (Borgatti & Everett, 2006). Cohesiveness is measured on walks, and specifically, the length of walks from length of one walk as calculated by the degree centrality metric through to the infinite walks of eigenvector centrality (Bonacich, 1987).

Borgatti (2005) describes graph flow as either:

- Transfer - the flow of a discrete thing between a pair of nodes (e.g. loaning a friend $1,000),
- Serial duplication – the flow of something in a serial way that can be passed through a graph that can change the state of a node upon contact (e.g. contagion),

- Parallel duplication – the flow of something in a parallel way that can be passed through a graph that can change the state of a node upon contact (e.g. broadcast),

Walks can be measured in terms of a radial walk which is a walk between source and target node (degree and eigenvalue), and a medial walk which counts the walks through a given vertex. Freeman's betweenness centrality is the best known metric that measures medial walks - the number of shortest paths through a given vertex (Freeman, 1977). The concept of count can capture the volume or length of walks. Volume is considered the total number of walks (e.g. degree centrality, eigenvector, betweenness) and length is the distance between a given vertex to the remaining vertices in graph (e.g. Freemans closeness centrality).

The limitations to centrality indices are central to their appropriate usage and interpretation. Fundamentally centrality metrics are a rank. They are designed to identify the most important nodes and largely due to the scale-free topology of most graphs does not generalize well to the "unimportant" nodes and are sensitive to network topology (Ghoshal & Barabási, 2011).

Degree centrality is the measurement of how many neighbours a node is connected to (see Figure 2.10. for an example). The notion is to identify those entities with the most neighbours. In directed graphs the out-degree (a count of all edges out of a node) and in-degree (the count of all edges into a node) are variants. Degree can be used to understand who the most prominent nodes are and also as an information quantity metric (Coles, 2001). It stands to reason that the more information we have on an individual or community the more likely that that localised higher level of complete graph is likely to be reflected as an artefact within the metric. This is critical to explicitly caveat and from a data quality perspective is useful as a proxy for information quantity. Centrality and brokerage are key concepts related to the ability of entities to influence and control the flow of information.

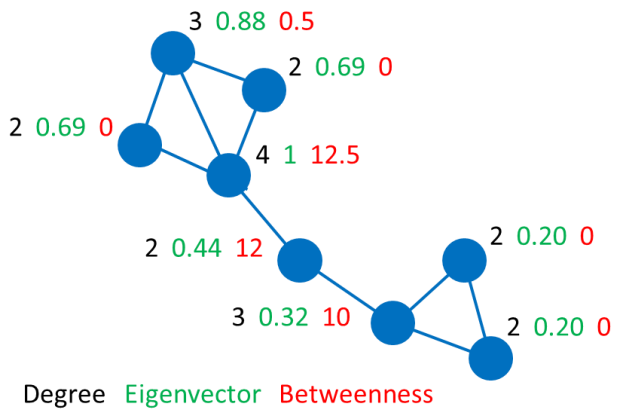


Figure 2.10. This figure illustrates degree centrality, eigenvector centrality, and Betweenness score for the toy graph.

Eigenvector centrality is an extension of degree and uses radial walks to measure how influential a node is (see Figure 2.10. for an example). The score is derived by iteratively summing each node's neighbours scores (beginning with each node neighbour equal to 1) and then generating a proportional score by dividing this score by the largest value in the graph, until the scores reach equilibrium. In this way "high-scoring" neighbours contribute more to a node's score than "low-scoring" neighbours. Eigenvector centrality is designed to identify those entities that are connected to highly connected or prominent, entities, which can be interpreted as measuring status. Katz centrality and PageRank can be seen as variants of the eigenvector centrality (Katz, 1953; Page, Brin, Motwani & Winograd, 1999).

Betweenness centrality identifies nodes that are relatively non-redundant (i.e. unique) bridges for communication between others (see Figure 2.10. for an example). This is done by counting the number of geodesics (shortest paths) that contain the node (Freeman, 1977).

Castells (1996), using metrics to measure information flow, then outlined how information drives the modern age and described how this information is propagated though the 'Network Society'. This empirically crystallised the importance of information flow in social contexts. Within the further constraints of criminal networks this information flow is even more prominent due to the clandestine nature of accessing the scarce knowledge and resources required to execute the tasks efficiently.

Centrality measures have been the first set of network measures applied to dark networks as they are simple to deploy and have high face validity. Centrality metrics have undoubtedly contributed toward the identification of central entities within criminal networks (Klerks, 2001; Krebs, 2002; Morselli, 2009; Morselli, 2010; Everton, 2013), however it is appropriate to acknowledge that these metrics have to be interpreted in the context of the data. This is borne out in the research on dark network resilience where the best strategies are those that include combining centrality metrics and roles (Bright, Greenhill & Levenkova, 2011). Often data in the criminal domain is collected in an ego-centric manner, thus any centrality metric will in part be an artefact of this data collection strategy.

Interestingly, Morselli (2009) states that the utility of using betweenness as a measurement of brokerage decreases when applied in increasingly large graphs. The basis of his argument being that as the length of geodesics being measured increases its relevance to the concept of a local broker consciously gaming his or her structural position decreases. This point is noted however, we would argue that as a complex system there will be many facets of observable behaviour, like where the immediate local goal of brokering, that from a wider perspective, perhaps driven by a self-organising process, may create unintended and directly unobservable benefits. An example of this could be the maturation of a broker that goes from controlling local transactions between

wholesale and retail phases on a specific commodity to controlling the majority of transactions between trafficking, wholesale, and retail phases and therefore wielding monopolistic power.

Rather than the simple interpretation of metrics in isolation it is useful to combine metrics. This can be done in multiple ways however using simple linear models to identify outliers is a common approach. Actors with high betweenness but low eigenvector centrality may be an important gatekeeper to a central actor, and an actor with low betweenness and high eigenvector centrality may have unique access to central actors. Morselli (2009) proposes that actors with low degree and high betweenness are concealed brokers versus a highly visible broker, and in fact he found the former associated to high ranking Hells Angels MC members and the later to low ranking Hells Angels MC members (Morselli, 2009). Calderoni (2011) found entities that were lower in degree centrality and high in betweenness centrality to be well strategically positioned to maintain control of criminal opportunities and manage risk. Unsurprisingly these same entities were also less redundant (Calderoni, 2011). The important element here is the generation of hypotheses that domain experts can test given their intimate knowledge of the problem and their ability to collect further data and test hypotheses.

Brokerage

Brokerage has evolved as a concept from the early research of Marsden (1982) and others with a mass of subsequent research empirically observing the influential brokerage of information and resource (see Galaskiewicz, 1979; Gould, 1989) and gaining an advantageous strategic position as a product of that (Taube, 2004). Brokerage in this sense is simply defined as where a node mediates the interaction between two alters. More formally this is exhibited by the existence of intransitive triples ($i \leftrightarrow j \leftrightarrow k$ and $i \nleftrightarrow k$) where a node (j) mediates the interaction. Gould and Fernandez (1989) extend this notion by defining numerous brokerage roles based on the pattern of community membership of the three nodes within the intransitive triple. Of course brokerage analysis in this sense is dependent on membership assigned to each node. Gould and Fernandez's (1989) typology of brokerage roles follow, noting that it is defined using an undirected graph perspective. The Coordinator broker (j) mediates interaction between two nodes from within his/her own community (see Figure 2.11.). Coordinators tend to be important for local cohesion. The Itinerant broker (j) mediates interaction between two nodes that are in the same community - a community that the broker does not belong (see Figure 2.11.). The itinerant role is likely to be unstable and transitory in nature, due to the close proximity of the alters (e.g. the broker is perhaps a past gang member) (Taube, 2004). The Gatekeeper / Representative broker (j) mediates interaction between a node within his/her community and a node outside of his/her community (see Figure 2.11.). The role of the Gatekeeper is to filter communications, and

insulate leaders from direct interaction with entities within the group and entities external to the group (Gould & Fernandez, 1989). Representatives have control over what information flows from inside the group to other groups. The Liaison broker (*j*) mediates interaction between two nodes from different communities, neither of which she/he belongs too (see Figure 2.11.). The liaison role is likely to be stable and therefore more likely to be persistent.

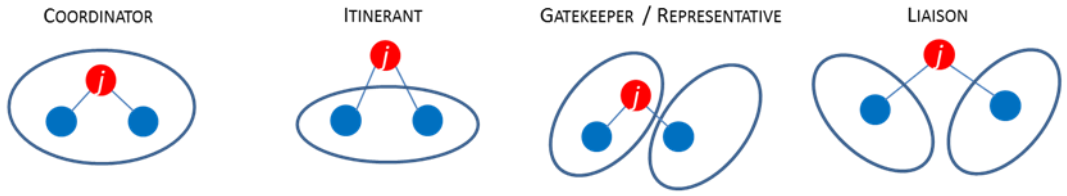


Figure 2.11. This figure illustrates the undirected versions of Coordinator, Itinerant, Gatekeeper/Representative, and Liaison.

The concept of brokerage is a direct measurement of an entity's ability to control the flow of information through their structural position. The inherent control of information derives influence (Turner, 1991), which generates an alternative set of hypotheses in terms of who is the most important set of entities to target, along with leaders and those high in the hierarchy (Coles, 2001). Furthermore, an understanding of brokerage gives a firm basis to understand what functional roles entities may engage in.

Measuring and understanding contextual brokerage opportunities will aid the identification of entrepreneurial entities that will likely manifest behaviour of playing their structural position in the network to advance their own goals through negative (e.g. stifling competition) and positive (e.g. connecting otherwise disjointed pairs) mechanisms. From a psychopathology perspective it is clear that the controlling opportunities that are created through being in a brokerage position are consistent with what is expected in the behaviour of a psychopath (Cleckley, 1988; Hare, 1999). To be explicit, it is likely that a psychopath would seek to create social position to exact manipulation and dominance, however only a fraction of brokers would display behaviours indicative of psychopathy. This is of contextual interest given the link between criminality and personality disorders (Hare, 1999). From a performance perspective there is research to indicate an association between brokerage and performance, and particularly where there is task ambiguity which is often the case in dark network contexts (Burt, 2004; Morselli & Tremblay, 2004).

The notion of criminal mentor extends and contextualises the notion of brokerage within the criminal setting. The role of criminal mentor was first posited by Sutherland (1937) with subsequent empirical research by Morselli, Tremblay and McCarthy (2006) substantiating this role and characterising the dyadic notion by criminal maturation, social capital, brokerage and degree centrality. Additionally, an

association has been found between brokerage and group leadership (Morselli & Roy, 2008; Morselli, 2009; Varese, 2013). These findings do typically generalize to larger more structured groups, however high brokerage tends to be a feature of mid-tier leadership rather than top level leadership (Calderoni, 2015).

Brokers within the "grey" sector also play an important role in facilitating crime and in particular organised crime (Reuter & Haaga, 1989; Jacobs & Peters, 2003; Morselli & Giguere, 2006; Amadore, 2007; Savona, 2012). Those lawyers, accountants, and complicit legitimate business people conduct brokerage roles in terms of laundering money, fundraising, registering corporate entities, provision of nominee services such as nominee directors, nominee shareholders, shell corporations, business addresses, creation of bank accounts (and specific correspondent banking accounts), wealth management advice, the recruitment of participants, and the provision of employment and accommodation (Morselli & Giguere, 2006; Morselli, 2010; FATF/OECD, 2010; Gimenez-Salinas Framis, 2014; Savona, 2012; Lo & Kwok, 2012; Leuprecht & Hall, 2014). It is clear from these findings that the scope of those legitimate actors' roles intersecting with the illicit sector is not just limited to status and expertise, but can creep into a more generalised brokerage facilitation role. Therefore, if we can identify businesses or brokers with specific skill sets (e.g. corporate registration) adjacent to members of organised crime groups we can attribute a higher probability that they are involved in the grey sector. Additionally, from a supply chain perspective brokers occupy these key structural positions mediating entities engaged in adjacent supply chain phases (Natarajan, 2006).

The involvement of legitimate actors in the grey sector is extremely influential from a criminal complex system perspective. And notably it is the weak ties that link these legitimate actors to criminal entities that fundamentally generates small-world topological properties enhancing the networks efficiency and effectiveness (Morselli, 2003, 2005, 2009). From this macroscopic perspective we can then start to identify specific industries, geographical areas or other conceptual partioning that has a higher relative incidence, or intersection, with criminal enterprise and so understand the drivers (e.g. cash businesses lead to opportunity to intermingle criminally derived profits aiding concealment of crime) of such phenomena.

Boundary Spanners

Related to the concepts of Brokerage is the notion of Boundary Spanners. Tushman (1977), amongst others, developed the idea that within innovation domains the set of nodes that span multiple communities and therefore spanning boundaries of communities, are a channel for innovation. This notion again overlaps with the concept that weak ties are responsible for access to scarce resources, including both information and functional resource (Granovetter, 1973). The

identification of weak ties has been deployed within Rodríguez's (2005) analysis of the Madrid terror network, who found that the "weak ties" were critical to the success of the network.

Local Transitivity

Transitivity (also known as the clustering coefficient) measures the probability that the adjacent vertices of a vertex are connected. Vertices scoring high in this metric are generally located in dense cliques or clusters – the building blocks of communities. Higher levels of transitivity are related to high cohesion and resilience, but generally lower efficiency (Simmel & Wolff, 1950). As such the measurement of local transitivity is useful in determining how topologically redundant a vertex is. Redundancy is a critical concept in terms of risk management and resilient networks. The ultimate goal is to target non-redundant nodes in the context of topology, role and position. Within the dark network literature an association has been found between non-redundancy and higher criminal earning (Morselli & Tremblay, 2004). These notions are related to structural holes, the empty spaces in the social structure that separate sources of novel information (Burt, 2004). These structural holes are related to weak ties (Granovetter, 1973) and indicate access to diverse information and control over information diffusion (Burt, 2004).

2.3.4 Mesoscopic and Macroscopic Knowledge Discovery

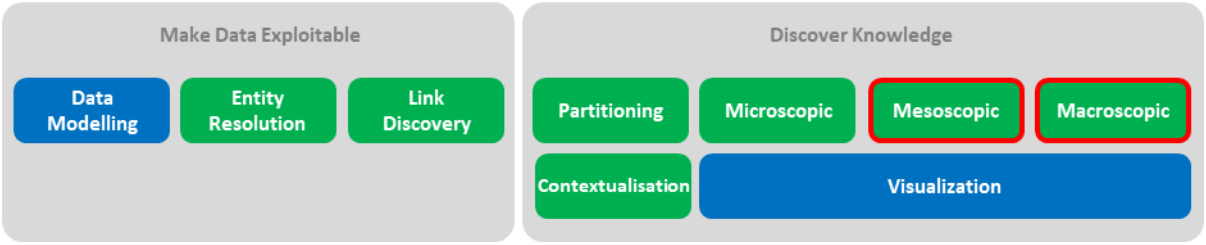


Figure 2.12. This figure outlines the modular design of GCND, with the current focus on mesoscopic & macroscopic knowledge discovery.

Mesoscopic knowledge discovery refers to the discovery of knowledge at the collective level in between the microscopic level (entity) and the macroscopic level (network). These levels can be described variously at the level of the dyad, triad, tetrad, cohesive subgroup, clique, clan, community, subgraph, etc. – we will use the term group to describe these. Macroscopic knowledge discovery refers to the discovery of knowledge in the context of the entire network. At the core of undertaking collective perspectives of how the complex system operates are the notions of emergence and self-organisation. These notions drive the modelling at the collective levels. Indeed, much of the focus at the collective level has been the topology of the network as a whole or as a group or subgraph level, and the mechanisms that lead to the topology that is manifest.

We will now briefly survey relevant topological features and the mechanisms that underpin the development of topology.

Scale-free networks

Scale-free networks are networks that exhibit a long-tailed degree distribution that approximately follows a power law (Barabási & Albert, 1999). That is a small number of nodes that have a very high degree and many nodes that have a relatively low degree. The scale-free nature of networks have now been observed in many networks (Albert & Barabási, 2002; Barabási, 2009) across a variety of network types including social networks (Jones & Handcock, 2003), metabolic networks (Jeong, Tombor, Albert, Oltvai, & Barabási, 2000) and the internet (Faloutsos, Faloutsos & Faloutsos, 1999). However, Broido and Clauset's (2018) examination of a range of real-world graphs found a diversity of degree structure, with only a relative few exhibiting significant scale-free structure. In particular, social networks were found to be generally not strongly scale-free, with around 58% only displaying weak scale-free structure.

Barabási and Albert (1999) proposed two mechanisms that drive scale-free topology: growth and preferential attachment. Growth being the incremental addition of a small number of nodes at each time step, and preferential attachment being characterised by those nodes that have a high degree (i.e. a lot of connections) receive a disproportionate number of new connections, compared to those nodes that have a relatively low degree. Scale-free networks thus will be characterised by localised star topology or hubs. Identifying groups and networks with a scale-free topology remains an important element to not only understanding the mechanisms causing scale-free topology but also provides context for understanding the impact that the topology of the network has on network resilience, maturation and also as a marker of data completeness and quality.

From a criminal domain perspective the actual topology of networks is somewhat masked by the partial data available to observe these networks, and the bias that exists through collection of data and the by-product of techniques employed to create a usable data asset. This only serves to indicate the importance of data quality assessment and highly accurate entity resolution and link discovery to get a clear reading of uncertainty. Hence, the importance of a computational solution like GCND.

Network Resilience

The resilience of a network is the property of a network that enables it to withstand node and / or edge removal and still operate at very similar levels as prior to the node / edge removal intervention. This property is perhaps best evinced by its application in epidemiology where the goal is to produce maximal effect (e.g. stop the spread of a disease) from an intervention (e.g. vaccination) against a

finite set of nodes. The goal is to understand which specific vaccination strategies lead to efficacious results. Interestingly, empirical evidence has demonstrated that scale-free networks are vulnerable to targeted attacks (Albert, Jeong & Barabási, 2000) of prominent nodes, with the efficiency of the networks inhibited significantly. But these same networks were found to be particularly resilient to random node removal. Similar findings have been repeated across a range of network types, including dark networks, with the most effective "attack strategies" consisting of methods that recalculate degree or betweenness subsequent to every node removal (Holme, Kim, Yoon, & Han, 2002; Carley, Lee, & Krackhardt, 2002; Carley, 2006; Morselli & Petit, 2007; Xu & Chen, 2008; Bakker, Raab, & Milward, 2012; Malm & Bichler, 2011; Bright & Delaney, 2013).

Resilience is not just a static quality but a dynamic one. Many networks, from a range of areas (e.g. neuroplasticity) are flexible and adapt to endogenous and exogenous shocks, displaying resilient characteristics. Criminal groups are no different in this regard (Carley, 2003; Carley, 2006; Morselli, 2009). Actors often adopt multiple roles, change roles and may be involved in a number of criminal ventures in parallel. Above all actors often display high redundancy and are therefore fairly easily replaceable. Understanding these microscopic features in the context of a highly embedded small-world scale-free graph and the pure economic demand and supply of criminality it is straight forward to understand how criminal networks are highly resilient to shock.

Xu and Chen (2008), in their examination of four dark networks (terrorist, criminal network, gang network and dark web), found that the dark networks all had scale-free and small-world qualities characterised by short average paths and high clustering coefficient, making them vulnerable to attacks on the bridges that connect communities rather than on hubs. An empirical study of an Australian methamphetamine trafficking network found that the most optimal intervention strategy to fragment the network into disjoint components was to target nodes based on degree and the entities role (Bright, Greenhill & Levenkova, 2011). Other strategies proposed include targeting entities with critical skill sets (Klerks, 2001), key low redundancy roles (Bright, Hughes & Chalmers, 2012), the key set of brokers (Morselli & Roy, 2008), and maximal fragmentation given set of entities size of k (Borgatti, 2006). Interestingly Duijin and Klerks (2014) found that targeting ring leaders had no or little impact of criminal networks. From a contextual point of view targeting specific phases of the supply chain is another potential strategy.

Carley (2003) found that targeting the most prominent entities was less successful in creating enduring network impairment than targeting emergent leaders. The resilience and regeneration of networks is a critical element and it is important to maximise the possibility that intervention impairs networks fundamentally rather than producing exogenous attacks that end up creating a higher operating group. Indeed, there is evidence, based on simulated models, that strategies need to be

contextual including elements such as recruitment and group maturation to ensure interventions are successful (Borgatti, 2003).

Therefore, the ability to understand the mesoscopic structure and topology of the criminal complex system gives critical understanding in terms of how to optimally supress criminal activity.

Assortativity

Assortativity, also known as homophily, is the notion that entities will be attracted to other entities with similar attributes, colloquially referred to in the phrase "birds of a feather flock together" (McPherson et al., 2001; Newman, 2003b), and is defined as:

$$r = \frac{1}{\sigma_q^2} \sum_{jk} jk(e_{jk} - q_j q_k) \qquad \text{Eq. (7)}$$

Assortativity of degree identifies how often entities with similar degree were connected within the network, and is thus a measurement of preferential attachment – a core driver of network evolution. Percolation is another feature of assortative networks that has been used to measure the robustness, or resilience, of networks. Callaway, Newman, Strogatz and Watts (2000) found that whether the community is assortative or not has important implications on its resilience to targeted and random attack. Assortative networks are much more susceptible to targeted attack, but resilient to random attack, meaning an intervention approach targeting central nodes will fragment the network and severely retard a networks ability to function.

Assortativity has also been applied to age in an attempt to understand, in particular, how homophily drives the evolution of networks and the entry of entities into social networks. Gaining a deeper understanding of these mechanisms will unlock our ability to predict missing relationships and predict network growth. As noted above Ugander, Karrer, Backstrom, and Marlow (2011) found facebook users had a marked preference for connecting with users of a similar age. The assortativity of age and gender is well documented within criminal networks (Reiss, 1988; Weerman, 2003; van Mastrigt & Carrington, 2013).

Small-world effect

The notion of a small-world is based on anyone in a network being able to communicate with anyone else within the network through only a very small number of intermediaries. The small-world effect was spectacularly brought to the attention of the world through the famous 1960's studies conducted by Stanley Milgram where he demonstrated that letters could be passed from an originating source through only a small number of intermediaries to finally get a successful delivery to the designated

recipient (Milgram, 1967). This demonstrated that a short path, of around length six, did indeed exist between any pair in the network, and provided empirical support to the small-world theories that Pool and Kochen (1978) were expressing. Watts (1999) built on these concepts of communication through short chains and described how widely social relationships influence not only our immediate neighbours but cascade through the "horizon" beyond this visible neighbourhood to friends of friends and beyond, contextualising the small-world effect. This process was then demonstrated by Christakis and Fowler (2007) on the diffusion of obesity, smoking, and alcoholic intake through social networks, which they generalised as the notion hyperdyadic diffusion.

From a graph theoretic standpoint the small-world phenomenon was categorized by Watts and Strogatz (1998) who noted that a small-world had to be characterised by being relatively large, sparse, decentralised so no one entity dominates the network, and highly clustered (or transitive). This notion of small-world was then formalised within the Watts and Strogatz model that is characterised by a small average shortest path between a pair of nodes and a large clustering coefficient (transitivity). The average path length measures the average of all shortest paths between pairs within the community, and conceptually measures how efficient the community is at exchanging information and resources. Clustering coefficient, or global transitivity, is a metric of cohesion which is based on the probability that the adjacent nodes of an ego node are connected. Empirical evidence suggests this metric is more robust than graph density (the ratio of the number of edges to the number of possible edges), particularly in terms of its sensitivity to community size. Cohesion is an important concept as it is hypothesized that communities that are very cohesive are inefficient in information flow and that communities that are not cohesive may have sub-optimal trust in members (White & Harary, 2001).

The features of a small average path length and high clustering coefficient have been noted in a range of networks including infrastructure, social, protein-protein interactions and ecological (see Boccaletti, Latora, Moreno, Chavez & Hwang, 2006). The small-world is a topological feature that enables measurement and inference based on the network context, particularly in the area of efficiency and network maturation. A considerable body of research has identified abnormal small-world properties in patients with dysfunctional brain function (Hsu, Wu, Cheng, Chen, Lu, Cho & Lin, 2012).

Interestingly, many studies focusing on drug networks, using a variety of data sources, have found scale-free and small-world properties (Brantingham, Ester, Frank, Glässer & Tayebi, 2011; Malm & Birchler, 2011; Xu & Chen, 2008). These features are thus critical to understand when developing a computational solution to construct micro, meso and macro understanding of the criminal complex system.

Centralization

Centralization is a measure of how central the network's most central node is in relation to how central all the other nodes are (Freeman, 1978), or in other words how much the network clusters around a few key nodes or is more decentralised or distributed. Centralization is implemented by an algorithm that calculates the sum in differences in centrality between the most central node in the community and all other nodes in the community, and then divides this quantity by the theoretically largest sum of differences in a same sized community. It is however critical to understand this within the context of the completeness and quality of data, as extremely centralised communities are unlikely. Therefore, this concept along with other topological features can all be used to assess data incompleteness, and factor this critical knowledge into the discovery of knowledge from the network in both a local and global sense. Furthermore, centralization can be used in combination with degree assortativity to better identify hierarchical communities that are not artefacts of the data.

Interestingly, Gimenez-Salinas Framis (2014) found low density (0.2 to 2%) and low centralization (17.3 to 21.3%) in three wholesale oriented groups that were sized at or about 60 participants, and one smaller group (23 participants) that displayed higher density (14%) and centralization (41.7%). Following, this it was found that the three groups displayed a more heterarchal structure with higher horizontal division of labour, and the other smaller group was hierarchically structured around a single leader. As centralised networks will have an approximate scale-free degree distribution, and as such they are proposed to be less resilient to targeted attack (Albert, Jeong & Barabási, 2000; Bakker, Raab & Milward, 2012).

Brokerage

The number and quality of brokers that a community engages with may be a useful measurement of how criminally sophisticated or mature that community is (Coles, 2001). The notion of broker can be measured by multiple metrics, including betweenness, Gould and Fernandez's brokerage, and weak ties.

Emergence

Emergence lies at the core of the complex systems view, and as such provide a critical block of GCND, evinced explicitly in the use of assortativity and network resilience, and more implicitly in the adoption of the graph perspective and the influence of actors and groups of actors on individual actors behaviour. Emergence was first explicitly demonstrated in a social context in 1969 when Thomas Schelling revealed how behaviour (racial segregation) at the macroscopic level, whilst driven by individual behaviour, is not exhibited at the microscopic level in individual behaviour. In other words

complex systems display emergent properties where properties measured at the collective do not quantitively reflect those properties as a sum of properties of the individual actors. The underlying concept is based on the idea that a dynamic complex interaction takes place between various microscopic and macroscopic levels of the system resulting in an emergent behaviour or property (Goldstein, 1999).

The concept of emergence thus has a natural fit with social psychology – and specifically human behaviour in relation to groups – which creates a strong inter-disciplinary theoretic and empirical basis. This basis can then form the conceptual foundation of a range of models deployed within GCND. Models that enable the discovery of latent knowledge of criminal behaviour across the microscopic macroscopic spectrum. Models that are highly dependent on quality data (e.g. individual behavioural metrics and relationships between entities), provenance, appropriate data representations (e.g. graph), and creative use of technology to mitigate uncertainty and enable the development of advanced complex system based models.

This theoretic and empirical basis is built on actor's participation in both local and non-local contexts, explicitly referring to an actors interaction with their immediate (local) observable neighbourhood generating a direct influence and the unobserved set of actors beyond the local neighbourhood that have indirect influence.

In 1958 Heider posited balance theory stating that balanced relations between neighbours produces harmony and thus perceived controllability and predictability, whilst imbalance creates instability, unpredictability and tension. Therefore actors manifest behaviour to reduce imbalance. Newcomb's symmetry theory (Newcomb, 1953) extends balance theory by placing in the context of dyadic communication rather than pure cognition, and importantly retains the core assumption of "persistent strain toward symmetry". If there are divergent views on a topic, and therefore a lack of symmetry, the strain toward symmetry will be dependent on the pairs individual "attractiveness" of the alter and the perceived importance and position of each alter on the topic. Newcomb states that the more an individual comes into contact with people he or she is attracted to and that hold differing views the more likely a person's views will change. This leads to notions of social influence. Seminal studies by Milgram (1963) were performed demonstrating how individuals behaviour can be controlled by perceived authority (e.g. leadership position) and tactics used to minimise or dampen conflicting cognitions (cognitive dissonance – see Festinger, 1956) leading to extreme behaviour (e.g. the perceived electrocution of people). In terms of an entities ability to influence (social power), French and Raven (1959) identified numerous sub-elements. Namely, reward, punishment (coercive social power), referent social power – where a reference group provides the basis for an individual to identify with (Kelman, 1961), expert social power – where an individual is influenced by another

individuals perceived expertise, informational social power – where an individual is influenced by another individuals perceived control of information as a resource, and legitimate social power – where an alters influence is derived from their position of authority. Asch (1951) demonstrated the degree to which individuals are influenced by real or imagined group pressure. The mechanisms are likely both explicitly via overt group behaviour (Asch, 1951) and implicitly via modelled behaviour (Bandura, 1971). In the context of social influence are the important areas or research in deindividuation and group polarization. Deindividuation refers to the reduction of a person's sense of personal responsibility and identity in the context of groups (Zimbardo, 1970). Group polarization is the notion that individuals adopt an extreme attitudinal position, based on the group's overall attitude position (Myers & Lamm, 1976). Interestingly, Lee (2007) found that deindividuation was indeed associated to stronger group polarization. This is an important finding in the context of the criminal domain as many behaviours of the key referent organised criminal groups are inherently designed to create deindividuation such as compulsory club attendance (Veno, 2002), wearing unwashed overt gang patches, compulsory riding on powerful motorcycles, the overt display of offensive insignia such as swastikas and white supremacist symbols (Montgomery, 1976; Smith & Fox, 2002), and imposing ritualistic deviant initiation rites (Montgomery, 1976). Within the context of this deindividuation is the collective deviant behaviour engaged by criminal actors, and the peripheral collective belief sets that from group to group differ. A clear example of extreme deviance is that provided by outlaw motorcycle gangs whose core value is to engage in narcissistic antisocial self-gratifying violent behaviour, often in the form of internecine violence and opportunistic violence, in addition to organised crime (Tretheway & Katz, 1998; Quinn, 2001; Smith & Fox, 2002). In addition to these important elements is the context that functional groups of entities engaging in criminal activity are goal-oriented, with actors' dependent on others to complete tasks and fulfil roles with uniformity.

From a graph or systems point of view, dependent on the quality of the data, behavioural concepts such as attitude, role, position, and power, and how these shift over time, can be measured (with a stated uncertainty) and combined with the relational perspective the graph provides to underpin the modelling of network and systems properties, including emergent properties. These concepts are central to understanding and quantifying group and individual behaviour within the criminal context. Such insight can enable a more advanced and nuanced response. For example, inferring how a functional criminal group operates from a structural and functional perspective is an important goal as it informs the relevant organisation (e.g. police) where a group is situated from a maturation point of view (e.g. a newly formed group going through a specific phase of development – see Tuckman, 1965), creates context for group weaknesses to exploit (e.g. a lack of peripheral cohesion with recent changes in membership may indicate a set of entities to target for informant information), and gives a structural context to understand entity roles and what intervention strategy is likely optimal to impair the network to a maximal extent.

2.4 Summary of Literature Review

The relevant literature spans a range of diverse disciplines including, graph theory, psychology, forensic psychopathology, management, genomics, sociology, statistics, complex systems, and computer science. These disciplines, in combination, provide the essential theoretical and empirical foundation, when applied through the criminal lens, to the development of GCND. The "low hanging fruit" problem was outlined and the building blocks of the solution offered. These building blocks consist of firstly taking a complex systems perspective, essentially viewing the criminal domain as not so much individual independent events but as an inter-related web of activity that has many dimensions of causal influence, particularly at a micro meso and macro level. The review followed in line with the sections of the computational solution developed, "Make Data Exploitable" and "Discover Knowledge". Within the "Make Data Exploitable" section data modelling and the value of representing data in graphs was discussed in the context of uncertainty and data incompleteness. Entity resolution, a fundamental core module of GCND, was a particular focus due to its importance to the effective functioning of the solution. Both pairwise and collective entity resolution was covered, with a particular graph focus. Link discovery encompassing both link inference and link prediction was also covered in depth as the second module of GCND focuses on generic link prediction. The combination of deploying accurate entity resolution and link prediction mitigates data incompleteness and uncertainty and enables a maximally exploitable dataset and creates the opportunity to discover latent knowledge. The discovery of knowledge is the second section to GCND and is premised on intertwining theoretic and empirical knowledge derived from the domain and discipline literature creating contextualised knowledge. Two elements emerge from the criminological and criminal network literature as high utility focal points – the supply chain and attitude. To extract meaningful reproducible knowledge leveraging the concepts of supply chain and attitude a range of generic concepts need to be applied. Generic concepts include community (using graph partitioning or community detection), equivalence (using blockmodeling), and brokerage are core. Many important additional measurable concepts are covered at both the micro and the meso/macro level to round off the literature review to provide the basis for understanding the current implementation and potential extensions of GCND, emphasising the importance of these concepts for the assessment of emergence – a core abstract component of complex systems. Part B will build on top of this foundation articulating the design and implementation of GCND.

Part B: Methodology, Data, Design and Implementation

Part B covers the methodology and data used to test the computational solution, and what and how the computational solution has been designed, unit tested and implemented in the context of the criminal domain. Part B has been divided into three chapters. The first chapter is "Evaluation methodology and data" which covers the methodology and the data from which the evaluation was based. The second chapter is "Make Data Exploitable", covering the modules entity resolution and link prediction, which are critical to optimising the quality of the data from which insight is to be derived, and particularly instances where the integration or federation of data is necessary. The third chapter is "Discover Knowledge" which focuses on how to discover latent knowledge, from a systems perspective, from this optimally resolved criminal focused graph, utilising a range of metrics across a variety of perspectives (micro, meso, and macro).

Evaluation methodology and data [chapter 3]

It is useful to describe the methodology and data used to evaluate the performance of the computational solution at this point as it illustrates in a clear concrete way the kind of data GCND has been designed to treat, how that data was protected from a security and privacy perspective, and the evaluation methodology adopted. The datasets have been selected as a heterogeneous collection of datasets that will provide a range of performance perspectives, enabling a more well-rounded evaluation.

3.1 Evaluation methodology

Creating the ability to rigorously and robustly evaluate the GCND solution and each unit that comprises GCND was critical. This was established by attempting to mimic applied real-world settings as closely as possible whilst balancing ethical, privacy and security aspects.

The four datasets were selected as the conceptually minimum datasets required to test GCND on the criminal domain. The four datasets focus on the conceptual constructs of risk / criminality, transactional, assets, and corporate vehicles (including corporate vehicles that utilise features to obscure the beneficial ownership, otherwise known as non-transparent vehicles), however in real

applied settings the expectation is that a broader set of datasets should be included to give a fuller view of the criminal system.

Having said this, the four evaluation datasets provide a good simple set of heterogenous datasets from which to evaluate the solution's performance across a range of variables. Most notably scalability is nicely contrasted in the range of datasets size.

Each dataset, where relevant, was used to test each significant unit of code, providing the ability to express the strengths and weaknesses of each unit and the modules and code overall. As a by-product of utilising the same evaluation sets throughout readers can develop a clear idea of what each dataset represents and gain a real incremental understanding of how the data is progressively made exploitable and latent knowledge discovered.

The data was stored and processed in its entirety within an appropriate New Zealand government secure facility and complied with New Zealand government's rules for the protection of classified information and code of conduct. The author maintained the appropriate New Zealand government clearance across the entire period of the thesis.

There is no disclosure of any personally identifiable data from the four evaluation datasets within this thesis. Any names used within examples and figures are fictitious and any resemblance to any current or past entity is merely coincidental. The absence of publishing any real entity details in conjunction with the inability to reconstitute any individual or group level identities from the content published removes the ethical risk in relation to protecting individual's rights. A secondary consideration however is prejudicing the NZ government in their ability to maintain the law. This risk was mitigated through ensuring the balance of value to public and private practitioners in combating criminal activity far outweighed any potential knowledge gleaned by criminal entities that can be used to obfuscate their digital footprint. In fact, an argument can be put forward that publicising the generic advanced methods used to detect crime generates a cost to the criminal system if criminal actors choose to respond to the threat (e.g. hide assets) and does not guarantee non-detection.

Transparency is an important element of evaluation. We attempted to use the simplest and meaningful metrics designed in a way that enables replication as easy as possible. So, where feasible units were not only tested in isolation to assess their performance but were then assessed in terms of their contextual value in terms of increasing the performance of the whole. The other benefit of relying on the simplest metrics is that it makes it somewhat easier and therefore hopefully clearer when conveying the actual performance of functions. Performance measures were always conducted in the context that over-fitting or coupling models to the data too closely will simply result in poorer generic application of the model, and so, little time was spent on tuning parameters to extract small

amounts of value. The context was firmly that the code was designed to deal with a range of situations and provide value without creating the requirement to spend a large amount of time on tuning every single function.

The key metrics used to evaluate module effectiveness are Accuracy, Precision, Recall, F-measure, and Cohen's Kappa Coefficient. This set of metrics was selected on the basis of their wide usage and simplicity, enabling broad comparison and clarity. The bias of using this set of metrics on imbalanced problems is significant and well documented (Powers, 2011), however presenting Precision, Recall, and F-measure together in the context of explicitly describing the level of skew mitigates the bias whilst retaining simplicity and transparency.

At times objective performance metrics are not available or possible, and so there is a reliance on a combination of the logical premise and face validity of the function and the use of subject matter experts in subjectively assessing performance. The use of subject matter experts is problematic as there is a firm reliance on the quality of knowledge possessed by the experts, and not just domain knowledge but broader ability to deal with concepts and visualisations they may not have previously been exposed to. Also, some of the concepts and knowledge generated within this paper is completely novel which requires a level of abstraction from experts in terms of judging the plausibility of results, thus exposing them more to bias. So, it is important to highlight the tentative nature of any performance assessment in this light.

All of the evaluation conducted was using RStudio 1.0.143 and R 3.4.2 on a Windows 10 environment with a CPU employing Intel Xeon @ 2.20GHz (8 cores) and 64 Gb RAM.

Let's now turn to the data used for evaluation purposes.

3.2 Sanctions data

Sanctions data comprises four underlying datasets sourced from the US (https://sanctionssearch.ofac.treas.gov/), UK (https://www.gov.uk/government/publications/financial-sanctions-consolidated-list-of-targets/consolidated-list-of-targets), EU (https://eeas.europa.eu/headquarters/headquarters-homepage_en/8442/Consolidated%20list%20of%20sanctions) and UN (https://www.un.org/sc/suborg/en/sanctions/un-sc-consolidated-list). Sanctions lists include "*individuals and companies owned or controlled by, or acting for or on behalf of, targeted countries"*, in addition to *"individuals, groups, and entities, such as terrorists and narcotics traffickers designated under programs that are not country-specific"* (US Office of Foreign Assets

Control) that pose economic, trade and national security risks. As such, the sanctions data provides a useful window to global entities of risk. The data creates an extreme case for entity resolution as the lists contain multiple aliases and transliteration variants for a range of person and corporate entities.

The data is open source and was extracted on the 8th August 2016 and was transformed into a harmonised property graph format. The raw graph consists of ~23,000 vertices and ~44,000 edges, including ~14,000 person entities, ~7,000 organisation entities, and ~800 addresses. Interestingly, the topology of the sanctions data is such that it is near complete, which has an impact on both ER and LP.

3.3 Dark Network and Suspicious Transactions (STR)

Dark Network and Suspicious Transactions data is NZ government criminal and transactional data. The data provides a great example of multiple sources of data curated through manual process, plus transactional data. There is significant rich text data that creates insight into the criminal history of many entities. This data contains a large number of complex and intentional name variation and therefore requires an ER approach that has high performance on this specific sub-problem.

The data was extracted on the 7th August 2018 and was transformed into a harmonised property graph format. The raw graph consists of ~360,000 vertices and ~900,000 edges, including ~100,000 person entities, ~15,000 organisation entities, and ~100,000 transactions.

3.4 Offshore Leaks

Offshore Leaks data is an umbrella term comprising a number of datasets including, Panama Papers, Paradise Papers, Bahamas Leaks, and Offshore Leaks (https://offshoreleaks.icij.org/). Each of these datasets was derived from separate leaks of data is made via the International Consortium of Investigative Journalists (ICIJ). The Paradise Papers was leaked from the offshore law firm Appleby and made available in 2017 and 2018; the Panama Papers was leaked from the Panama law firm Mossack Fonseca and made available in 2016; the Bahamas Leaks was leaked from the official corporate registry of the Bahamas and made available in 2016; and the 2013 original Offshore Leaks data was sourced from two offshore service providers Portcullis Trustnet and Commonwealth Trust Limited. These datasets in combination provide a unique glimpse into the domain of offshore corporate entities. The data is focused around global corporate relationships from a small number of sources and as such the presence of supernodes (highly connected nodes) is common – although all of the supernodes are entity resolved, unlike the NZ Companies Office data. Much of the data was manually annotated from documents which led to significant typographic error, and a sparsity of

attributes. For example, there is a significant number of unknown entity types and instances where person entities names have not been parsed, introducing more error when it comes to ER.

ICIJ have invested much resource to clean and transform the data into a reasonable state for use. The open source data was extracted on the 19th January 2018 and was transformed into a harmonised property graph format. The raw graph consists of ~1.4 million vertices and ~2.4 million edges, including ~400,000 person entities, ~640,000 organisation entities, ~250,000 address entities, and ~80,000 where the entity type is unknown. In terms of the topology of the Offshore Leaks data we observe a very low global transitivity and disassortative degree – markers of a star and hub topology.

3.5 NZ Companies Office

The NZ Companies Office (NZCO) is a register (https://www.nzbn.govt.nz/using-the-nzbn/nzbn-services#bulk-data) of all limited liability companies registered in NZ, comprising all registered companies, their directors and shareholders, with addresses. A point in time extract was collected on the 28th August 2018. The open source data provides a complete picture of companies registered in NZ, and provides a "low signal" challenge in terms of entity resolution as entity type is often unspecified and person's names are often contained in a single string rather being parsed into atomic elements of family and given names. There is significant data error as many fields within the register are not validated. Interestingly the NZCO data contains a large proportion of duplicate entities and unduplicated supernodes.

The raw graph consists of ~18 million vertices and ~93 million edges, including ~7 million person entities, ~4 million organisation entities, ~3 million address entities, ~ 2 million sundry entities, and ~2 million where the entity type is unknown. The NZCO data has bipartite tree-like graph features where the person class of nodes does not have intra-edges. There are however edges from companies to companies and the infrequent directed cycle. This unique graph model and resultant topology has a significant impact on ER and LP.

See table 3.1. for a summary of metrics which helps describe and quantify each dataset, both as a raw dataset and the data subsequent to entity resolution.

Table 3.1. Outlines key descriptive metrics of each dataset used for evaluation.

Profile of data sources (before and after ER)

	Sanctions		Dark Network / STR		Offshore Leaks		NZ Companies Office	
	Pre ER	Post ER	Pre ER	Post ER	Pre ER	Post ER	Pre ER	Post ER
Vertices	~23,000	~15,000	~360,000	~280,000	~1.4 m	~1.2 m	~16 m	~8 m
Edges	~44,000	~44,000	~900,000	~900,000	~2.4m	~2.4m	~90 m	~90 m
Persons	~14,000	~8,000	~100,000	~50,000	~400,000	~300,000	~7 m	~2.5 m
Organisations	~7,000	~5,000	~15,000	~10,000	~640,000	~570,000	~4 m	~1.5 m
Addresses	~800	~800	~235,000	~30,000	~250,000	~230,000	~3 m	~2 m
Transactions	0	0	~100,000	~100,000	0	0	0	0
Unknown entity type	0	0	0	0	~80,000	~80,000	~2 m	~400,000
Degree Assortativity	0.9416	0.8405	-0.0167	0.1981	-0.0445	-0.0502	-0.059	0.6240
Mean degree | Max degree	7 | 200	12 | 2,200	4 | 3,500	7 | 28,000	3 | 37,000	4 | 37,000	11 | 98,000	4 | 192,000
Global transitivity	0.9794	0.9772	0.0148	0.0838	0.0004	0.0006	0.0014	0.0131
Mean Shortest Path	1.9047	1.9659	8.0528	19.613	17.79*	16.62*	14.706*	12.525*
Diameter	8	8	30	86	26*	28*	25*	20*
Small world Quotient	3,439	1,228	1,035	3,767	173*	177*	2,143*	5,998*
Centralization – degree	0.0049	0.0736	0.0047	0.0507	0.0266	0.0315	0.0030	0.0109

* Metrics estimated via a sampling approach of 1% of pairs.

The range of metrics give a good quantifiable basis for a basic topological understanding of each dataset, particularly in the context of ER and LP. The size and distribution across entity types is self-evident.

The connection, cohesion, and small-world nature of the graph is best measured by global transitivity, diameter, average path length, and small-world quotient (higher score relates to a smaller world) determining how dense clusters are and how connected these clusters are across the graph (Amaral et al., 2000). The sanctions data is extremely clustered with very high global transitivity and low mean shortest path (MSP). The other three datasets have low global transitivity (approaching tree like numbers) and much longer MSP. Data that is not entity resolved at all will have a global transitivity close to zero (a tree), as every entity is essentially unique. The NZ Companies Office data and Offshore Leaks are examples of this.

The degree distribution is a significant feature of each dataset, with the mean and max degree giving an indication, not only of how scale-free the distribution is, but also that the entities with the highest degree are those most likely to have a high number of duplicate entities. This feature is very important in the context of ER due to the pairwise intractability problem.

In regards to the number of duplicate entities existing in each dataset it is straightforward to assess, given the change pre and post entity resolution, however of course the same real-world entity may have many duplicates. So this number does not reflect the number of unique real-world entities with duplicates.

Make Data Exploitable [chapter 4]

"Make Data Exploitable" focuses on taking raw input data that represents the criminal problem and transforming this raw data into a data representation optimised to apply a complex systems perspective. The raw data is typically comprised of a range of tables with inconsistent data models. Each raw dataset is transformed into a target generic property graph model. This is aimed at harmonising data elements so as to retain as much data richness as possible. Subsequent to the multiple datasets being represented in the required generic property graph format they can be represented as a disjoint graph ready for data fusion using entity resolution. Following entity resolution, where duplicated real-world entities will be identified and resolved, link prediction is applied to predict real-world relationships that are not represented in the data. These steps are crucial to ensure the data is optimally represented for knowledge discovery. Figure 4.1 represents where "Make Data Exploitable" sits in the context of GCND.

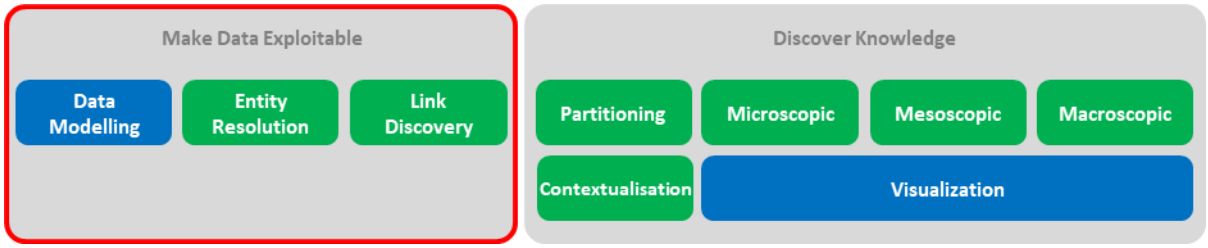


Figure 4.1. This figure outlines the modular design of GCND, with the current focus on the "Make Data Exploitable" section.

4.1 Entity Resolution

Criminal networks - graph representations focusing on criminal actors - present significant challenges in terms of deriving an accurate representation that mimics real-world reality. Incompleteness, data heterogeneity, non-intentional error, intentional misinformation, and bias all contribute to increase the uncertainty of the data. At the core of this uncertainty and variance is accurately and reliably resolving duplicate entities which in fact represent the same real-world entity (Benjelloun et al., 2009) - entity resolution (ER).

Whether the problem is the integration of multiple heterogeneous datasets or focuses on the entity resolution of one homogeneous dataset, the specific complexity of the criminal domain places particular demands on an entity resolution solution. This complexity can be driven from artefacts of the source(s) of data and their representation or the wider domain where data error is generated from both incidental and purposeful intentional provision of misinformation. Interestingly within the

criminal domain the very entities that are the source of intentionally poor quality data are often the very entities that are of most interest.

The criminal context provides an additional layer of complexity and uncertainty due to the motivation of entities to actively supply misinformation with the goal to reduce the effectiveness of entity resolution. For this reason entity resolution and link discovery are often deployed in concert to enhance the quality of the graph through making the data as explicit as possible and discover latent knowledge. This section however is limited to entity resolution. A critical element though to highlight is that entity resolution in the criminal domain must be able to contend with not just missing nodes and edges, but the existence of fake and spoof nodes. Fake nodes are nodes that are in the dataset but do not exist in the real-world and spoof nodes are instances where a real-world node will be represented as multiple nodes within the dataset (Maeno, 2009). The high uncertainty of criminal data caused by error, high incompleteness, and the presence of fake and spoof nodes (and edges) can lead to poor performing ER contributing to an obfuscated graph.

Therefore, inexpensive, accurate and scalable approaches to ER that go beyond identifying the obvious matches (deduplication) and can also detect the non-obvious matches are of critical importance. Non-obvious matches refer to the set of matches that are not exact matches, but include variation. Current "state of the art" commercial entity resolution products are often focused on markets that require generic scalable fast deduplication solutions and do not place the requisite emphasis on the detection of the complex low-signal non-obvious matches. However, when dealing with domains that are marked by complexity, rare instances, and high cost of failing to identify duplicate entities - terrorism, organised crime, terrorist financing, and complex tax crime - an ER solution is required that targets non-obvious duplicate entities accurately, at enough speed and scalability to ensure the solution can be deployed pragmatically. Responding to this need the Entity Resolution module has been developed to support the detection of non-obvious duplicate entities.

To achieve this goal, remembering that "there is no such thing as a free lunch", would require a significant shift in approach. To do otherwise would at best result in making some improvements at the margins, and more likely generate a solution inferior to those more mature and resource-rich solutions on the market.

Entity Resolution module

The design of the Entity Resolution module is fundamentally a semi-supervised learning model (see Figure 4.2.). The computational solution focuses firstly on engineering a broad set of relevant contextual features across as close to all relevant pairs as possible, first using the augmented target data and then secondly using the contextual prediction data (including all features extracted). Then

after generating contextual features we subsequently make a decision, using machine learning, on which pairs are the same real-world entity.

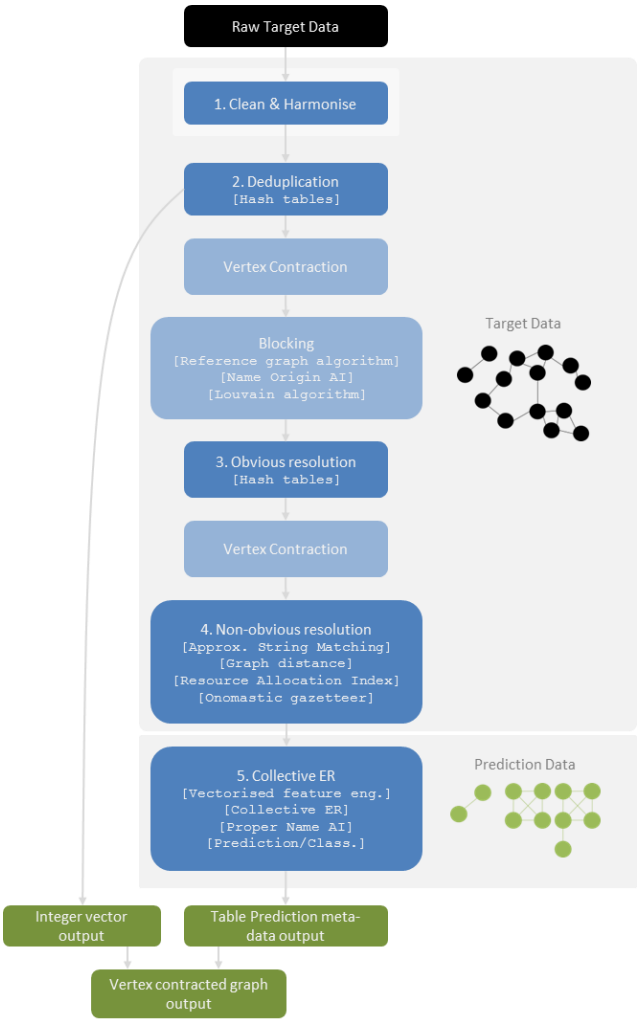


Figure 4.2. Modular design of the Entity Resolution module.

The model is implemented by breaking feature engineering into five sub-modules. The goal is to explicitly improve the data as we go, reduce the data size, reduce the problem space, and concurrently create more relevant features to enable better decisions – particularly on those pair instances that have high uncertainty. High uncertainty can be derived from a combination of lack of data points available and complexity. The five sub-modules are:

1. Pre-processing – Error handling, data cleansing, data harmonisation, and the generation of vertex attribute metadata,
2. Deduplication – focusing on those entities that are exact matches,
3. Obvious Resolution – focusing on those entities that are near exact matches – often derived from typographical and transcriptional error,

4. Non-Obvious Resolution – focusing on complex cases, often derived from transliteration, significant name change, system artefacts (e.g. name order, use of initials, etc), and intentional differences,
5. Collective Entity Resolution – using the data generated thus far – prediction data – to start to 'close the world' and create contextual metadata to enable better decisions.

Let's now briefly cover the five sub-modules and describe the output that is generated. This will create enough context to explore a number of fundamental problems that required solving. This investigation will answer some of the readers' questions around why certain design decisions were made, and will serve to highlight the key novel elements of the Entity Resolution module. Subsequent to this we will cover the performance of the model which should provide an objective basis from which to judge whether the design decisions were astute.

4.1.1 Sub-module 1: Pre-processing (error handling, data cleansing, harmonisation, and generation of metadata)

Data cleansing and harmonisation is all about ensuring the data is consistently represented across the dataset, without the significant loss of information, independent of operating system. Elements deployed here include explicitly encoding character data as UTF-8, altering all text to upper case, and removal of special characters. This is an incredibly important step, particularly in data derived from unstructured sources, and when integrating heterogeneous sources of data. Of course these steps reduce the information available for decision-making (e.g. changing all text to upper case may reduce the information available for name parsing), however these fundamental aspects should be addressed prior to ER, reducing the impact of information loss. The data input to sub-module one is a table of vertices with attributes:

- 'Name.1' – first given name, if relevant [character],
- 'Name.2' – second given name, if relevant [character],
- 'Name.3' – remainder of given names, if relevant [character],
- 'Family.Name'– family name, if relevant [character],
- 'DOB' – date of birth, if relevant [character, dd-mm-yyyy],
- 'id_Label' – the label of the entity (e.g. the name of a company, the digits of a phone number, the full name of a person) [character],
- 'id' – the identification number for each vertex [integer],
- 'Semantic_Type' – the type of entity (e.g. person, organisation, phone, email address) [character],

- ‘Provenance’ – the source of the data [character],
- ‘Date’ – the date the data was created [character, dd-mm-yyyy].

and a table of edges with attributes:

- ‘Source’ – the id of the vertex that is the source of the relationship [integer],
- ‘Target’ - the id of the vertex that is the target of the relationship [integer],
- ‘Semantic_Type’ – the type of edge (e.g. ‘associate of’, ‘shareholder of’, ‘address of’) [character],
- ‘Provenance’ – the source of the data [character],
- ‘Date’– the date the data was created [character, dd-mm-yyyy].

or an igraph object with the same set of vertex and edge attributes.

The second component of this sub-module is the generation of vertex metadata to support subsequent sub-modules. Metadata generated includes approximate age, suffix identification, country tagging, domicile inference, name frequency, community detection (Louvain algorithm), blocking algorithms (metaphone3, Reference Graph, label truncation), proper name origin classification, proper name classification, and hypocorism graph.

Approximate age identifies the approximate age of each person entity that has a date of birth. Ages that are illogical (for example a year of birth in the future) are simply overwritten as NA. Representing date of birth data in a numeric format enables a variety of efficient ways to measure age pairwise similarity.

Suffix identification identifies those entities containing suffixes and either harmonises or removes the suffix and creates additional metadata to indicate the use of that specific suffix. For example, person entities using the suffix of “Junior”, “Jnr”, “Senior” or “Snr” needs to be identified and treated appropriately to support decision-making. Corporate entities with a suffix of “LTD” is simply harmonised to “LIMITED”. The treatment of suffixes is an important step to make data as harmonised and explicit as possible.

Country tagging is the process of identifying the respective country for each address. A regular expression approach is used. The approach takes each country in turn and identifies a set of addresses that contain specific regular expressions (based on country name, country abbreviation, high frequency locations such as cities, provinces, states). The set is then hierarchically pruned by identifying addresses that also exist in other higher frequency sets and do not have an explicit country noted. For example, the address “1 Queen St, Christchurch” could be tagged as both “New Zealand”

and “Barbados”, so we choose the tag that has a higher frequency, indicating which country tag is more likely. For address entity resolution this additional metadata can be extremely useful for making more efficient and better decisions. The dataset of countries, cities, towns, regions was manually curated through using Wikipedia as the sole source (https://en.wikipedia.org/).

Domicile inference uses the country tagging as an input and through graph inference tags person and corporate entities based on their adjacency to country tagged addresses. This approach allows for a person or corporate entity to be domiciled in multiple jurisdictions. Geographical based data enables decision-making to be conducted within a constrained or bounded context (Boundedness). In circumstances where there is little data available, or the decision is complex, having additional geographical context can be crucial. For example, knowing that a pair of entities both have a relationship with addresses in Luxembourg and Bolivia can provide that extra context enabling successful resolution. This feature is incredibly important when attempting to resolve corporate entities from a global perspective, as a common corporate name (e.g. “Offshore Investments Ltd”) can be used across multiple jurisdictions.

Name frequency [numeric, range 0-1] is a normalised Bayesian numeric measure where 0 is unique and 1 is the most common name. The Name Frequency Algorithm is deployed as a computationally efficient heuristic approach that accepts atomic names as independent for computational performance reasons (although they are clearly not) and measures how unique they are relative to all other atomic names. Importantly the algorithm is originally seeded by taking only original full proper names (i.e. the family and given names of a person) so not to generate bias created due to duplicate names. For example, within the Offshore Leaks data the atomic name “Gaetanne” is noted as a name on 28 occasions, however these instances all relate to the same person and so is counted as 1. Name frequency creates a metric that provides direct stochastic context on the likelihood of a positive match, in combination with a range of equivalence metrics and other metadata. For example, undertaking ER in an English speaking country the pairs of person entities “Jane Anderson | Jane Anderson” and “Gigi de Paolo | Gigi de Paolo” whilst displaying similar amount of information are likely to be manually ER assessed by a human very differently. A significant contributor to that human decision-making is the concept of name frequency. Humans can quickly establish and contrast how often they have heard names before and therefore utilise this knowledge to make better decisions.

Community detection (using the Louvain algorithm) is the graph partitioning approach discussed earlier that is used to partition sets of nodes into localised groups. Community membership is another metric that creates social context on the distance between entities. As a graph-based metric it is clear that as the quality of the overall graph improves so does the quality of the community detection. A

core application of community detection is as a blocking algorithm creating an orthogonal approach to blocking algorithms based on string-based data.

Blocking algorithms in general are a standard approach to deal with pairwise intractability. Within the Entity Resolution module numerous algorithms are used to both maximise the chance to capture close to all relevant pairs, but also to enable very specific application of pairwise based metrics because of the context created by the blocking algorithm. For example, using the unique ordered letters algorithm (e.g. "David Robinson" == "ABDINORSV") may generate sets of names which used in combination with the Cosine ASM identifies instances of atomic name transposition (e.g. "Robinson David" and "David Robinson"). Blocking algorithms used, or available to be used, include metaphone3, label truncation, community detection (Louvain), unique ordered letters, and Reference Graph Algorithm. The Reference Graph Algorithm is a key novel component to the successful performance of the Entity Resolution module and will be covered in detail within the next section.

The Proper Name Origin Classifier is a meta-blocking algorithm designed to predict the origin of a person's name. This enables more specific and granular blocking, resulting in less operations and increased specificity. This classifier is also used to identify and resolve instances of anglicisation. This classifier will be covered in more detail in the next section.

The Proper Name Classifier is an algorithm designed to identify instances of where pairs of person entities may have similar names – as measured by the ASM algorithm – but are not likely to refer to the same real-world as they both contain proper names that are similar but differ (e.g. "Ken" and "Ben"). This is distinct to a pair of entities that have a similar name where one has a proper name and one is not a proper name but contains a typographical error (e.g. "Ken" and "Kwn"). For example, the names "Norma" and "Norman" are very similar using a range of ASM algorithms, however it is clear that they are likely referring to two different people. This classifier will be covered in more detail in the next section.

The Onomastic/Hypocorism graph is used to identify instances where different names are used interchangeably, often in the form of nicknames, and diminutives (for example, Mikhail & Misha or Benjamin & Ben). The graph is a directed simple non-weighted graph constructed from an edgelist table that has been manually constructed and curated and includes Anglo, Russian, Spanish, Arab, and Farsi pairs (for example, Edward & Ed; Dmitri & Dimitry; Mohammed & Mohd).

The graph is partitioned by component to support a simple targeted blocking approach, and the edgelist itself is used as a lookup.

The output of sub-module one is an augmented igraph object, with the same set of attributes as the input.

4.1.2 Sub-module 2: Deduplication

Deduplication is focused on the efficient identification of nodes that are exact matches and contain enough information that ensures they refer to the same real-world entity. The definition of what equates to "enough information" is completely domain specific and so this is left to the user to define as a parameter. The technology underpinning the efficient computation in this sub-module is hash tables. The input to this sub-module is the igraph object outputted from sub-module one, and the output from this sub-module is an efficient membership vector, where each entity is assigned the same membership as those that are deemed exact duplicates, and a contracted graph based on the input igraph object and the membership vector generated. The rationale for this separate sub-module will be covered in more detail in the next section.

4.1.3 Sub-module 3 and 4: Obvious and Non-Obvious Resolution

Obvious and Non-Obvious Resolution sub-modules utilise the same code framework (see Appendix A for details of code mechanics) but have differing goals. Both sub-modules use a pairwise equivalence approach with the Obvious Resolution sub-module targeting pairs of entities that are exact or close to exact duplicates, relaxing the information required to make a decision compared to the Deduplication sub-module but still yielding high certainty matches. These pairs are often derived from typographical error.

The Non-Obvious Resolution sub-module makes use of the same framework and wrapper function as the Obvious Resolution sub-module but uses a series of metadata to go beyond the obvious and uncover latent knowledge to enable detecting equivalent pairs where there is a lack of data available (e.g. A B Smith equivalent to A B Smith?) and / or significant differences in the attributes of the entities (e.g. A Smith-Brown equivalent to A Smith?). It is important to point out that at this stage the data has been contracted twice subsequent to the Deduplication and Obvious Resolution sub-modules, ensuring the size of the data is minimised and the quality of the data is optimal, ensuring graph metrics are as accurate as possible.

The pairwise equivalence framework consists of applying a specific *wrapper function* multiple times with varying parameter settings targeting various specific subsets of entities and scenarios.

The wrapper function is comprised of an *indexing function*, an *equivalence assessment function,* and a *decision-making function.* The *indexing function* determines the set of entities to focus on (e.g. all

Persons with Chinese origin names that have at least a family name, a given name, and a date of birth). The *equivalence assessment function* uses a blocking strategy to divide the set into blocks and applies an ASM algorithm in parallel over all blocks (e.g. use the Reference Graph Algorithm to block in conjunction with computing string similarity via the Jaro-Winkler ASM algorithm) returning a set of pairs that score over the threshold supplied. The *decision-making function* takes this set of pairs and their ASM scores and applies a range of operations to enable an accurate decision on whether to accept the pair as a validated pair or not. Operations include using the hypocorism graph to determine the use of a nickname, using community, graph distance, or RAI to determine social distance, using domicile inference to determine geographic distance, using name frequency to determine how common the pairs name is, and using date of birth and age operations to determine age equivalence.

The output of deploying each *wrapper function* includes a table of pairs with supplementary metadata, dependent on parameters set, at both the pair level and the function level. The output graph from sub-module two is used as the input into sub-module three, and as the code progresses through sub-module three and four the wrapper function is deployed a number of times. The metadata from each applied wrapper function (a table of ER predictions and associated meta-data) is then collected and used as an input (known as the 'Prediction Data') by the Collective ER sub-module.

Note that there is the initial set of ASM thresholds used to determine the boundary of which pairs to retain, and then a more restrictive set of ASM based thresholds, in combination with other pairwise metadata, used to identify which pairs are invalidated. Both validated and invalidated pairs are retained. This is an important feature as the aim is to capture metadata about all possible pairs that represent the same real-world entity, so we can use the next sub-module to make the best contextual decision possible.

4.1.4 Sub-module 5: Collective ER

Collective ER consists of vectorised feature engineering, collective ER, proper name classification, and prediction/classification. The Collective ER sub-module takes the table output ('Prediction Data') from the Obvious Resolution and Non-Obvious Resolution sub-module and builds a graph consisting of all of the predictions, both validated and invalidated, with all available metadata – let's call this the Prediction Graph. This Prediction Graph then serves as the basis from which to conduct;

- vector-based approaches to help validate and invalidate potential matches derived from the Prediction Graph,
- contextual transitive closure on non-transitive clusters of the Prediction Graph to identify latent matches,

- contextual exclusivity to invalidate potential matches so logic holds, and
- prediction and / or classification using machine learning.

The specific vector-based approaches include a series of functions that target specific scenarios that are difficult to target in a pairwise approach. The three deployed functions target those person entities that have given name initials, transposition of names, and those person entities that have names from an Arabic origin.

Contextual transitive closure is conducted using "localised" transitivity and all of the metadata available. "Non-localised" transitivity is generated by simply using the pairwise wrapper function to measure the similarity between each entity within the cluster in a very specific way using the graph of metadata available.

Contextual exclusivity is conducted by taking the augmented Prediction Graph generated from the addition of edges via contextual transitive closure and measuring the new transitivity of each cluster and each edge. Any edges that have source and or target nodes that are less than completely transitive are then contextually reassessed in conjunction with a range of relevant metadata (attributes such as ASM equivalence, tuple distance, social distance, name frequency, information available to make a decision, name origin, initial similarity, prefix/suffixes, and nickname).

Boundedness, a measurement of how bounded the search space is, is then measured and a threshold generated via a distributional approach, influenced by the Tolerance parameter, to use as a method to identify potential matches that are extremely common coupled to minimal data available to make a good decision.

A radial based SVM (support vector machine) or RPART (recursive partitioning) algorithm then takes the table of prediction metadata (derived from the Prediction Graph) and performs a binary classification and probability assessment using the training data (set at default of the smaller of 100,000 or 90% of the prediction pool) generated through the model, using the validation feature to provide the observations. This generates an alternative classification and a more granular probabilistic metric from which to make decisions on which predictions to apply to the original data. It is important to point out that the machine learning algorithm takes in both pairwise metadata and the provenance of how that metadata was generated to give a contextual assessment. The rationale for providing the option of either SVM or RPART is based on the user's context of how urgently the results are required and how much accuracy is required, given that RPART is fast and less accurate and SVM is slower and more accurate.

The output from the set of sub-modules include a membership vector (from sub-module two) and a table of prediction metadata. The table of prediction data can be used as a knowledge asset itself to inform decisions, or it can be used to persist "same as" links within the original data, or used as a basis to contract (merge) the vertices into a more accurate graph representation.

4.1.5 Output

The following outputs are produced.

1. Membership vector generated from the Deduplication sub-module,
2. A table of predictions, derived from the remainder of the sub-module, with associated metadata including:
    - String Distance [numeric, range 0-1],
    - Social Distance [integer, range 0-2],
    - Name Frequency [numeric, range 0-1],
    - ER_Rule [character]: the specific ER functions that generated the metadata,
    - Local Transitivity [numeric, range 0-1],
    - Information Quantity [numeric, range 0-1],
    - Frequency [integer]: the number of functions that predict the pair are equivalent,
    - Uncertainty [numeric, range 0-1]: the uncertainty score (see above),
    - Validation [integer, range 0-1]: whether the Entity Resolution model predicts the match is valid or not,
    - Reason [character]: rationale for the validation decision
    - SVM_Classification [integer, range 0-1]: the binary classification of the SVM,
    - SVM_Prediction [numeric, range 0-1]: the probabilistic likelihood of the SVM.

Provision of the membership vector and the prediction table provides the basis for the user to link or contract the original graph-based on the edge list and metadata supplied, using models predictions or using the metadata to generate a new set of predictions.

3. A contracted graph, in the same format as the input graph, is also generated based on 1. and 2.
4. A diagnostics file is generated that covers the following for each component of the model and overall:
    - The number of predicted matches [integer scalar],
    - The proportion of unique predicted matches, or in other words the proportion of matches predicted by this wrapper function alone [numeric scalar],
    - The global transitivity of the predicted matches, an indicator of how the wrapper targets latent matches, and includes false positives [numeric scalar, range 0-1],

- The mean information quantity, a quantitative measure of how conservative or aggressive each wrapper function predicts matches, given the amount of information available [numeric scalar, range 0-1],
- The mean Name Frequency, reflecting partially how much information is available (person entities with fewer words in their name will generally be more common), but also explicitly gives a measure of how relatively common the matches are [numeric scalar, range 0-1],
- Uncertainty, giving an explicit measure of how accurate each wrapper functions matching performance was [numeric scalar, range 0-1],
- Number of invalidations, gives the sum number of how many matching predictions were invalidated through name negation, sub ASM threshold, exclusivity, or sub uncertainty threshold [integer scalar],
- Runtime (seconds) [integer scalar].

And, additionally for the ER model the following metadata is recorded (for demonstration purposes Offshore Leaks figures are included in parentheses):

- The number of entities contracted per type, for example the number of addresses (24,208), organisations (65,065) and persons (42,477) contracted [integer],
- The runtime (seconds) [integer scalar],
- The Tolerance parameter (e.g. 0.2) [numeric scalar, range 0-1],
- The Uncertainty threshold applied (e.g. 0.18) [numeric scalar, range 0-1],
- The Boundedness threshold applied (e.g. 0.37) [numeric scalar, range 0-1],
- The date and time of completion [character, dd-mm-yyyy hh:mm:ss],
- The amount of memory used (bytes) (e.g. 15,669,165,168 bytes) [integer scalar],
- The global transitivity pre and post Collective Equivalence Resolution sub-module (e.g. 0.9183, 0.9974) of the validated predictions [numeric scalar, range 0-1], and
- The mean diameter ratio [numeric scalar, range 0-1] of non-transitive components, with being perfect transitivity and closer to zero being non-transitive.

These metrics can then be used within an experimental framework to optimise the Tolerance parameter, and internal settings. Global transitivity and mean diameter ratio provide automated feedback on ER performance.

One would expect a global transitivity score of 1 if the matching was perfect and zero if it was random. The reason being that if *i* is equivalent to *j* and *j* is equivalent to *k* then logically *k* must be equivalent to *i* – a transitive graph structure. Global transitivity is measured once after the Non-

Obvious Resolution sub-module (pre) and once after the Collective Equivalence Resolution sub-module (post) to understand performance. Transitivity is particularly useful when comparing ER model performance. Mean diameter ratio focuses on the diameter of non-transitive prediction components providing a metric [0-1] where 1 equates to all prediction components being perfectly transitive, and as the score moves away from 1 so does the incidence of larger diameter non-transitive prediction components.

5. A graph-based visualisation of a sample of predictions in html.

The output generated for testing the accuracy of the model is in graph form, derived from a sample of potential matches. The method by default randomly selects 20 validated and 20 invalidated matches, extracts the source and target nodes and in an iterative neighbourhood approach identifies all incident edges to an order of *k*, currently set at 3. The goal is to select a representational group of matches, that is balanced (to potentially include TP, FP and FN) and large enough to be statistically useful, with enough context for the tester to make accurate verification. However, it is important to note that the 20 invalidated matches will present a biased picture of overall accuracy as this approach identifies 20 clusters of the most difficult to predict entities. Another way to explain this is that 80 nodes were randomly selected from the Prediction Graph as seeds to retrieve all known relevant prediction information. 40 of the nodes were randomly selected from the pool of nodes that were explicitly predicted to be the 'same as' another node, and 40 of the nodes were randomly selected from the pool of nodes that were explicitly predicted to be 'not same as' another node.

This approach creates visibility in terms of how the model determines which pairs are 'same as' and which pairs are 'not same as', which is critical so the users can not only give feedback on accuracy but also creates transparency enabling users to contribute to the development of the model by giving expert feedback. The drawback is that using this output for measuring the performance of the ER model will generate an overly conservative performance metric. This is important to highlight as the measurement of ER performance is subject to bias and the pragmatic difficulty of getting human testers to verify accurately and representationally across true positives, false positives, and false negatives – an inherently tricky imbalanced pairwise problem.

Figure 4.3. illustrates the graph output produced for users. The key details include:

- The layout algorithm used is "graphopt" (www.schmuhl.org/graphopt/) which has proved to be the most generalizable layout algorithm of graphs in the 1,000 to 5,000 node size.
- The file format generated is html, utilising the D3 JavaScript library, presenting the predictions in a dynamic graph format enabling testers to zoom and re-position vertices, which mitigates the issue of overlapping nodes, edges and labels.

- The graph is simplified to remove complexity, so multi-edges are contracted into a single edge. The complexity is much reduced making the tester's verification easier but modifying the ER models output potentially generates misconceptions.
- Each node has a label comprised of a concatenation of the entity's id and full label, to enable traceability and readability. Each edge label is comprised of three parts; Uncertainty [0-1], Social Distance [0,1,2], and Validity [0-1] which is the proportion of predictions the pair is a match. The edge also has a colour; grey edges are valid matches (i.e. Validity is > 0, or in other words at least one prediction that the pair is a match has been validated by the model), magenta edges refers to those invalidated matches that were invalidated due to the Proper Name Classifier – the source and target node contain real name words that are different –, pink edges refer to those instances of invalidated edges that were invalidated for other reasons such as exclusivity, under the ASM threshold, or under the Uncertainty threshold. The decision to limit the number of colours was based on keeping the visualisation as simple as possible for users.
- Metadata is provided on:
    - the sample graph provided giving the number of nodes and edges, breaking down the number of 'same as' predictions and 'not same as' predictions, enabling the Precision, Recall and F-measure to be calculated subsequent to human testing.
    - model parameters set or generated including Tolerance (default = 0.2), and Boundedness applied and Uncertainty applied, which refer to two probabilistically thresholds generated by the model.
    - date and time of the model deployment and the security caveat.

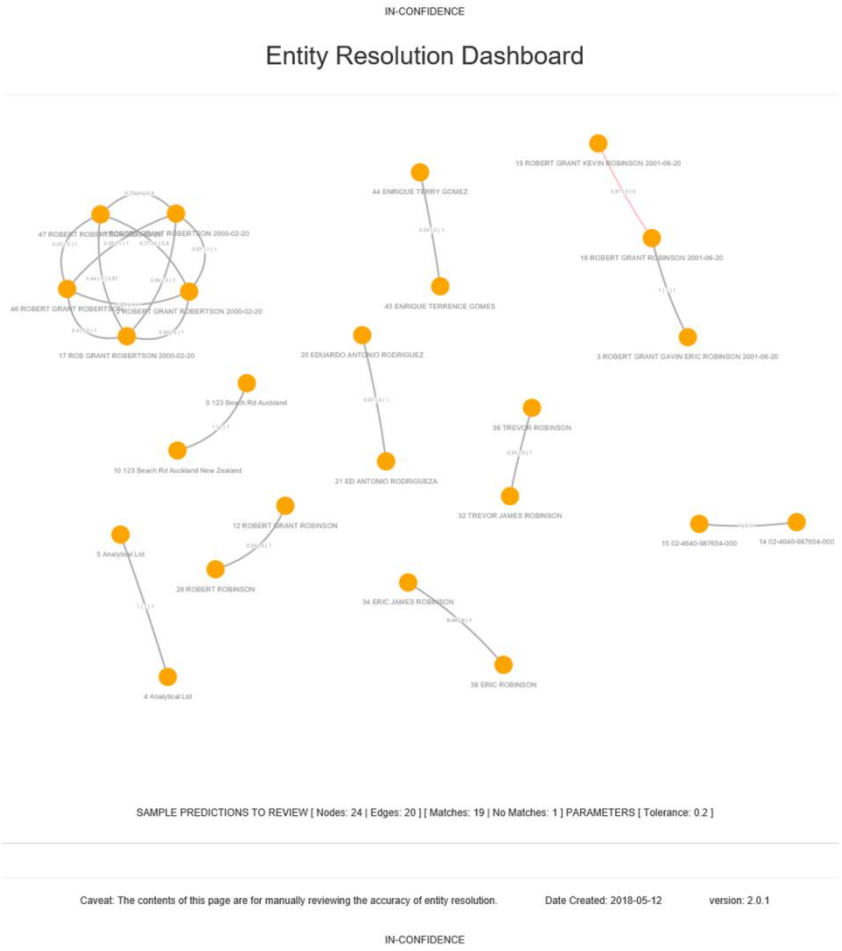


Figure 4.3. This figure illustrates a small example of the visualisation generated for testers (using fictitious data).

The sample graph is also generated for user testing that gives the tester a contextual basis for validating the performance of the model when assessing what parameters are optimal. The upper pane of figure 4.4. compares the pre and post global transitivity of the entity resolution across a range of Tolerance parameter settings, giving visibility over the model's generalisability, and in conjunction with the lower pane, performance. The lower pane relies on the input of human testers who manually verified a sample of potential matches. The proportion of entity contractions is also given for comparison. For the Offshore Leaks dataset it can be seen that from both figures that a lower Tolerance parameter is preferable. It is critical here to note that sampling and human error is significant and so minor fluctuations need to be seen in this context.

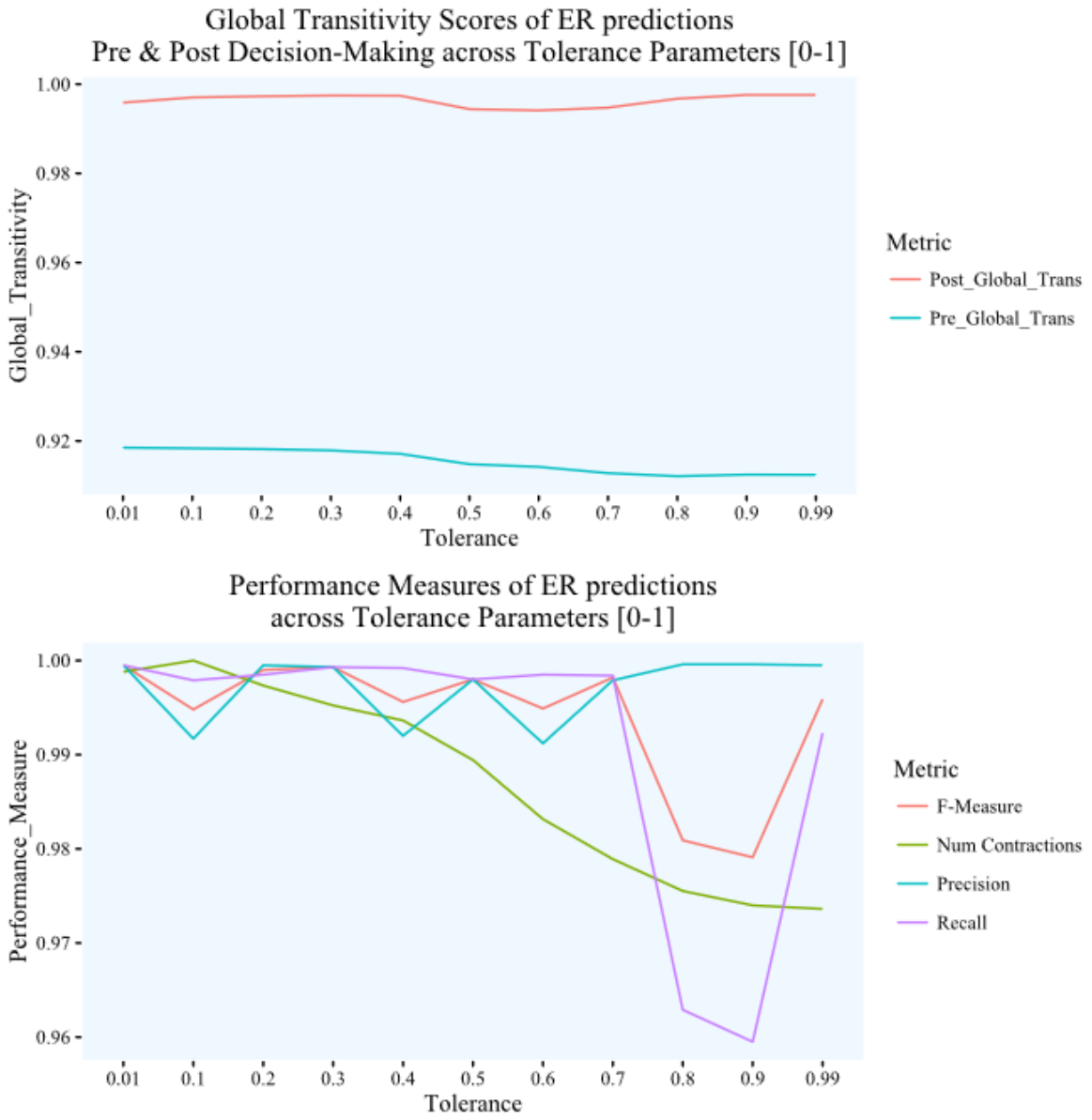


Figure 4.4. This figure visualises model metadata for the entity resolution of the Offshore Leaks. The upper pane compares the pre and post global transitivity of the matching across a range of Tolerance parameter settings. The lower pane contrasts accuracy measures manually verified from a sample of potential matches, across a range of Tolerance parameter settings.

I will now outline some fundamental sub-problems experienced with multiple representation of real-world entities in data and the allied sub-solutions. This will allow a targeted examination of key solution elements, and highlight the core features.

4.1.6 Entity resolution problems to address

Problem 1: Scale-free distribution of duplicates. It is common for datasets to contain many records in relation to a few entities and a few records for the majority of entities. See figure 4.5. below demonstrating the frequency of duplicates for each actual real-world entity in the NZ Companies Office data. Ignoring the 6,624,600 entities (~44% of original unduplicated graph) that had no known duplicates, and using a logged x axis to more easily represent the distribution of duplicate frequencies, we can see that the distribution is significantly skewed with a small number of entities having thousands of duplicates, with one entity having 11,019 predicted duplicates.

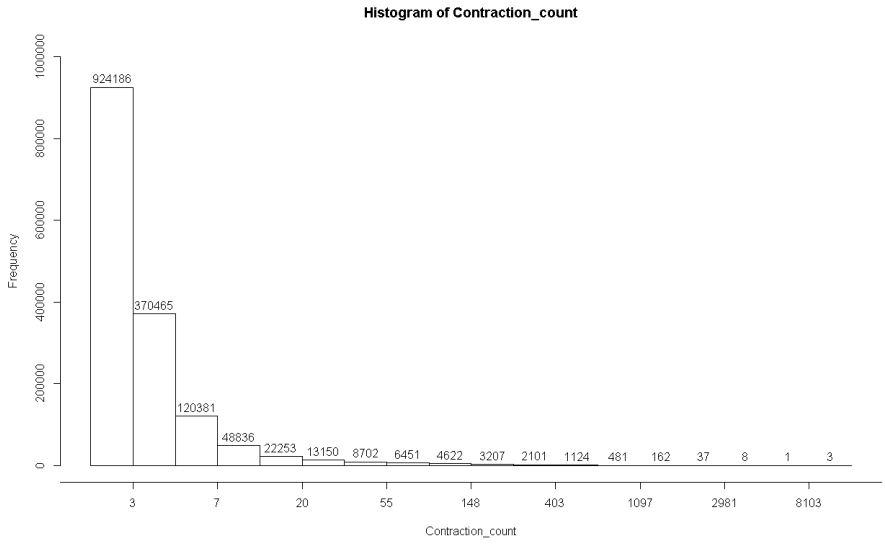


Figure 4.5. A histogram illustrating the frequency of entity duplicates (contraction count) in the NZ Companies Office data.

This topological characteristic highlights the pairwise intractability problem and renders any pairwise approach inefficient at the minimum and in extreme cases just not possible. The solution implemented is to perform a hash table lookup on those entities where there is sufficient metadata available to safely assess they are the same real-world entity. The output of this process being a membership vector indicating which entities are in fact exact duplicates. This task is referred to as deduplication.

Importantly, this membership vector is then used to conduct vertex contraction. The cost to contracting the graph is the time required to conduct the contraction. The benefits include the data decreases in size enabling faster subsequent computation time and the data is unequivocally more accurate enabling better quality metrics. For example, graph distance between entities or the commonality of proper names.

Problem 2: Community detection performance in the presence of supernodes. The inability of community detection algorithms to detect granular communities, particularly in localised areas around supernodes – vertices with an exceptionally large number of connections – is a well-known limitation of modularity optimization based approaches (Fortunato, 2010). This limitation is known as the resolution limit. Hence, we have to be careful when using community detection approaches to ensure that partitions identified are reasonable. Communities identified may include entities that have no indirect tangible relationship to one another. For example, many entities may transact with a casino, however a community that centres around the casino and the volume of its direct relationships will erroneously include all entities transacting with the casino within the same community. The consequence of this on community detection are the presence of very large communities that do not represent a community in the real-world. This presents real problems when using notions of graph distance to provide context on a pair of entities that potentially represent the same real-world entity.

The goal is to develop a generalizable partitioning approach that efficiently generates more accurate communities than current native applications, whilst retaining runtime efficiency. Yielding more accurate communities when blocking will enable comparing a higher proportion of relevant pairs leading to higher quality features and better predictions.

The proposed solution, refrains from relying on semantic context and metadata in relation to the vertices and edges, as not all datasets have the requisite datamodel and attribute richness to enable using the data in this way. The alternative solution focuses solely on the graph structure and simply filtering disassortative edges resulting in an improved graph structure.

This is done by firstly taking the absolute value of the source nodes degree minus the target nodes degree, with a score of zero being perfectly assortative edge, or in other words a connection between two entities who have the same degree. Then, secondly, by determining the change point in this absolute degree assortativity distribution and pruning those edges displaying significant disassortativity. The premise being that extreme disassortative (degree) relationships are more unlikely to involve enduring tangible relationships.

Problem 3: Pairwise intractability. The intractability of pairwise approaches is well documented, and is a problem linked to the approximate scale-free distribution of duplicates. The use of blocking algorithms to generate sub-sets from which to conduct pairwise comparison is the standard approach. Various blocking algorithms have been developed that have a range of associated cost and benefits. Algorithms include the phonetic based metaphone family of algorithms, simply selecting a set of letters from the string (e.g. the first letter, third letter, and seventh letter – “Robinson” == “RBO”), and a range of proprietary algorithms. Experiments conducted quickly concluded that whilst many blocking algorithms were computationally efficient the accuracy was a severe limiting factor to the

overall accuracy of the ER model. In response to this the Reference Graph Algorithm was developed. This blocking algorithm is relatively expensive to run (~ 11% of runtime) however it yields impressive accuracy, and when used in combination with other orthogonal blocking approaches, proves incredibly successful. Other blocking approaches used include community detection (Louvain algorithm), and ordered unique characters algorithm (e.g. "Robinson" == "BINORS").

Meta-blocking is another common approach used to split or combine blocks together to improve blocking performance. Within the Entity Resolution module the Proper Name Origin Classifier (PNOC) was developed to classify the origin of names. This enables the ER solution to not only speed up pairwise computation because of more sophisticated blocking but also tailor elements of the ER pairwise approach to specific sub-domains resulting in enhanced accuracy. For example, the PNOC can specifically target blocks of names with a Chinese origin versus blocks with an Arabic origin, thus creating the opportunity to efficiently utilise contextual semantic knowledge about that specific sub-domain, such as name characteristics and the source of name ambiguity.

Proper name origin metadata also creates additional contextual knowledge for decision-making, such as determining when a given or family name has potentially been changed through marriage or anglicisation. For example, when we are presented with two similar names - "Chandra Ravi" and "Chan Sandra" - we can use the PNOC ("SC", "CNEN") to avoid comparing names from differing origins which, due to their name similarity, might have otherwise been grouped together. In other words, a blocking algorithm may place both names in the same block (e.g. "CHA"), but the use of the PNOC as a meta-blocking approach can provide more granular blocks (e.g. "CHASC", "CHACN"). Additionally, due to the compound class "CNEN" we have determined that the name "Chan Sandra" is likely to include an anglicised atomic name, which is critical knowledge when making contextual transitive and exclusivity decisions in the Collective ER sub-module.

Blocking and meta-blocking is deployed with both the Obvious Resolution (2) and the Non-Obvious Resolution (3) sub-modules, enabling efficient pairwise metadata generation.

See the next section for more information on the Reference Graph Algorithm and the PNOC.

Problem 4: Absence of data. There will always be situations where there is not enough data available to make a conclusive decision. Approximate string matching (ASM) will not provide enough metadata alone to accurately make decisions with certainty. So, to minimise the uncertainty generated through lack of data the ER solution includes many other relevant features that are used to enable more conclusive decision-making.

Modelling the data in a way that highlights the relationships between entities provides us the ability to determine the context of how close a pair of entities are from a graph sense. This can be done using a range of metrics such as community detection, graph distance, and path distance. The distance between a pair creates decision-making context. For example, if a pair of entities with the common name of "Mark Smith" was identified, the uncertainty would be very high as there are many entities with this name, however if we then identified that they shared an address, then the likelihood they are the same person increases. However, there are some significant barriers to generating accurate and computationally inexpensive graph metrics. We will quickly cover these now.

Graph-based operations performed in native graph formats can be expensive, in terms of RAM expenditure and runtime. The solution to this problem for community detection, whilst retaining accuracy quality, is to modify the representation to a simple graph (i.e. making the graph undirected and removing multiple edges, loops, and weighted attributes) and using the Louvain algorithm. The following table (see table 4.1) illustrates the runtime differences between three high performance community detection algorithms; the Fast Greedy (Clauset et al., 2004), Label Propagation (Raghavan, Albert & Kumara, 2007), and Louvain (Blondel et al., 2008) algorithms.

Table 4.1. Outlines the computational expense (in seconds) of three community detection algorithms on both multi-edge and simple graph representations of Dark Network and NZ Companies Office data.

Computational expense (runtime in sec)

	Multi-edge graph	Simple graph
Dark Network STR data (~360,000 vertices, ~900,000 edges)		
Fast Greedy algorithm	NA	359
Label propagation algorithm	31	27
Louvain algorithm	8	10
NZ Companies Office data (~16 million vertices, ~90 million edges)		
Fast Greedy algorithm	NA	-
Label propagation algorithm	15,346	16,932
Louvain algorithm	1,989	1,144

From these results you can see that scalability of community detection algorithms is a challenge. The performance of the Louvain algorithm is clearly superior in terms of runtime, especially on a simplified graph. The accuracy of community detection algorithms is somewhat subjective, however experimental testing within the context of ER identifies that the Louvain algorithm provides consistent accurate results relative to other algorithms.

Limitations of scalability, runtime and RAM expenditure can be even more pronounced in other graph metrics. Graph distance (shortest path length) is a core element to various graph-based metrics (e.g. including the path-based RAI), which is particularly susceptible. The solution to this issue is the

development of a ‘table’ based - rather than a native graph - approach, enabling both parallel and distributed computing (e.g. Apache Spark) implementation. The graph distance function implemented will generate a score out to a path length of 5. As per the results in table 4.2, testing the graph distance function on a range of scale-free synthetic graphs, the benefits of implementing graph distance in a table-based approach, and particularly when using a distributed computing version, is clear. Scalability, runtime, and RAM issues are significantly alleviated when compared to a native igraph implementation in R. However, the topology of the graph is a significant factor, as calculating the Cartesian product is a core step within this algorithm. Hence, further testing needs to be performed to ensure performance holds across a range of graph topologies.

Table 4.2. Outlines the computational expense (in seconds) of graph distance (i.e. length of shortest path) on a range of synthetic scale-free graph datasets on four differing compute contexts.

Computational expense (runtime in sec)

Graph size (number of vertices)	R [igraph]	R [data.table]	Spark R [20 clusters @ 7Gb]	Spark R [56 clusters @ 3Gb]
10,000	4	2	61	
100,000	123	13	133	
200,000	396	26	134	
400,000	1,338	52	137	
800,000	5,313	104	137	
1.6 million		217	190	
3.2 million		445	206	
6.4 million			423	
12.8 million			791	291
25.6 million				442
51.2 million				742
102.4 million				1,745

The more resolved (contracted) the graph is the more accurate graph-based metrics will be, hence the raw graph is contracted twice within the ER solution - the first time after the deduplication stage, and then again after the obvious pairwise stage. Again the computational expense of vertex contraction can be prohibitive. Reimplementation of the native igraph vertex contraction function using data.table data representation reduces the runtime from 1,090 to 225 seconds on the STR/Dark Network and from 8,728 to 4,592 seconds on the NZ Companies Office data respectively.

A range of non-graph-based metrics are also relevant to ER. As mentioned above the commonality of names is an important concept to measure as it creates additional context for decision-making. A Bayesian function has been developed to measure each proper names uniqueness – generating a probability [0-1].

The amount of data and metadata available to make a decision is measured creating the context for the assessment of uncertainty. Metadata on the performance of feature engineering functions is also retained. This gives us the ability to take the performance of the feature engineering into account.

If address data is provided the model uses a regex dictionary approach to tag each relevant address as per its country, and then uses this country tag to ascertain what country each person and organisation entity has a direct relationship with. This creates additional geographical context for decision-making.

These concrete feature engineering approaches provide important metadata to ally more traditional features used in ER. From a more abstract perspective the Collective ER (4) stage is designed to begin "closing the world" creating significant opportunities to generate better contextual decisions.

Collective ER changes the focus of analysis from the raw data, or versions evolved from the raw data, to the table of predictions ('Prediction Data') and associated metadata derived from earlier stages. Doing this enables a number of things. Firstly, a key automated performance indicator can be derived easily providing a simple logical benchmark from which to both assess global accuracy and support local decision-making. This metric is transitivity, and is a key element to collective entity resolution (Bhattayachara & Getoor, 2009). The logic being that if *i* is equivalent to *j* and *j* is equivalent to *k* then *i* must be equivalent to *k*. Secondly, the problem space is now reduced. No longer do we make the assumption that the problem is open, we start to close the problem firstly to the entities we have a prediction about, and then we limit the scope of analysis even further to non-transitive components found within the Prediction Graph. Doing this provides two things - the ability to conduct more expensive operations to create metadata on more complex potential duplicates, and create context to making a new set of features relevant.

Vectorisation is used to engineer a variety of additional features, focusing on narrow parts of the problem such as name transposition and the use of initials. These procedures enable generating better quality equivalence metrics for each prediction.

Subsequent to the vectorisation approach the predictions and associated metadata are modelled as a graph (Prediction Graph). All predictions are assessed at this point using all of the metadata available with each prediction classified as valid [1] or invalid [0]. At this point all non-transitive components (with invalid edges excluded) are then examined. The logical foundation of this decision is that those non-transitive components are not logically sound and some uncertainty remains.

Transitivity is then employed. Each non-existing edge within each non-transitive component is then examined using a range of metrics including approximate string matching algorithms Cosine, Jaccard, Jaro-Winkler, and Longest Common Substring (Needleman & Wunsch, 1970), and graph distance. A tuple based approach is also used for name transposition, used in combination with an onomastic gazetteer to identify name transposition and the use of nicknames, and thirdly, used in combination with the name origin model to identify instances where it is likely an anglicised name has been used.

This additional metadata is then used to logically determine whether these new predictions are valid or not.

Having now exhausted attempts to identify new predictions via contextual transitive closure, we then employ exclusivity to identify what predictions cannot logically be accurate and then select the best alternative, based on a pruning strategy. Exclusivity is based on the premise that if *i* is not equivalent to *k* then *j* is not equivalent to *k* and/or *i* is not equivalent to *k*.

Exclusivity is implemented by identifying those edges that have a local transitivity below the change point of the distribution and the proper name is uncommon (assessed by the ~98th percentile score – dependent on the tolerance parameter). Edges that are retained also include those that have a graph distance of 1 (i.e. the source and target nodes share a community).

Boundedness is the concept of how bounded the problem space is when determining whether a pair is equivalent. This is critical as we take an open world stance – assume there is data that we do not have access to. For example, if humans are making a decision on whether a pair of entities are equivalent they will take into account how unique the entities names are, the geographical context of the decision – such as whether the data comes from a global context or a more restricted context such as a country, state, city or town – including where the pair of entities are domiciled, and how close they are in a social network perspective. These three concepts, name frequency, geographical distance, and social distance, are critical to establishing the context in which the decision is being made. Boundedness is important to apply hand in hand with the amount of pairwise information that is available. In other words when there is a wealth of pairwise information available (e.g. Family name, three given names, date of birth (DoB), address, adjacent actors in the network) boundedness can be relaxed, but when there is a dearth of available pairwise information (e.g. Family name and initials of two given names) then boundedness needs to be applied. Failure to get this balance between boundedness and information available exposes predictions to probabilistic failure, which is then reflected in lower global transitivity and mean diameter scores. So, when boundedness is set too high and pairs with little data available are considered equivalent then we observe a growing number of non-transitive subgraphs with a high diameter. Hence, we can use these topological measures and the distribution of metrics underpinning boundedness to optimise the application of the boundedness metric in the context of how much pairwise data is available. Methods to improve optimisation, both in terms of accuracy and speed, are ongoing.

Problem 5: The use of nicknames will not often be identified using ASM. The solution implemented is the classic onomastic gazetteer approach. The onomastic gazetteer is used to identify any person entities that could potentially be using an alternate name. For example, “Kevin” and “Kev” are equivalent onomastically so if a person entity has either name then it will be included within pairwise

equivalence assessment. The onomastic gazetteer was manually curated through using Wikipedia as the sole source (https://en.wikipedia.org/).

Problem 6: Some names are similar, when measured by standard approximate string matching techniques, but are actually different proper names. An example of this is "Bryan" and "Ryan". The solution to this problem is the deployment of the Proper Name Classifier. This model takes all atomic names used in the data, and the Prediction Data and conducts a binary classification as to whether that name is a proper name or a name generated through error. The output of this is then used to invalidate any prediction that contains differing proper names (e.g. "Sandy Brown" and "Andy Brown").

Problem 7: For purposes of auditability and transparency it is important to ensure as much of the metadata that underpins a prediction is made available to the user in a comprehensible way. Provision of complete meta-data for all predictions on a pairwise basis is inefficient and can be prohibitive computationally. The solution to this is a pragmatic one. The initial deduplication step involves exact matching and is represented merely as an integer vector, with entities that share the same number being assessed as duplicates. There is no explicit pairwise metadata provided for the deduplication, as this is inefficient and of little value. However, the output derived from steps 2 through to 4 is in the form of a table of predictions and associated metadata. Additionally, the membership vector and predictions can be used to contract the original input graph. Specifically, original id's are retained in all instances giving full auditability and provenance.

Problem 8: Many ER solutions do not provide a simple way to assess performance. This leads to multiple experiments, manual testing, and relies on the expertise of the user to find a somewhat optimal tuned model. The solution to this problem is based on a range of elements. Firstly, a diagnostics file is generated that provides detailed metadata about memory use, runtimes, parameter settings, sub-function performance, mean diameter and global transitivity. Two key automated metrics that provide unique insight into performance are mean diameter and global transitivity. Mean diameter refers to the mean diameter of non-transitive prediction subgraphs. Non-transitive prediction subgraphs contain error, whether false positive or false negative, and so measuring the diameter of these subgraphs gives a sound metric on quantifying the amount of error contained (see Figure 4.6.). Specifically, the mean diameter metric is the number of prediction non-transitive subgraphs divided by the sum of the diameter of those prediction non-transitive subgraphs, generating a 0-1 score, with scores closer to 1 interpreted as more accurate.

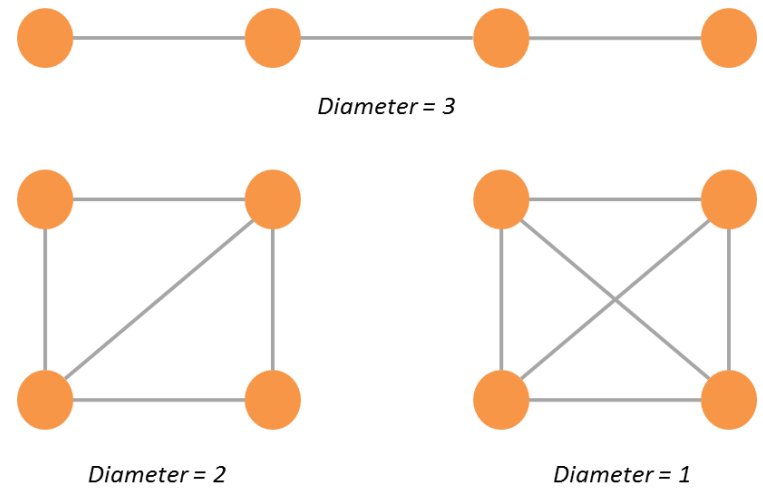


Figure 4.6. An illustration of the diameter metric on three subgraphs.

Global transitivity is very important as a logically sound performance metric, providing a firm basis to assess the Precision of the model, but not the Recall. The global transitivity measurement is generated by taking all prediction subgraphs and dividing the total number of closed triplets (see Figure 4.7.) by the total number of open and closed triplets (Luce & Perry, 1949).

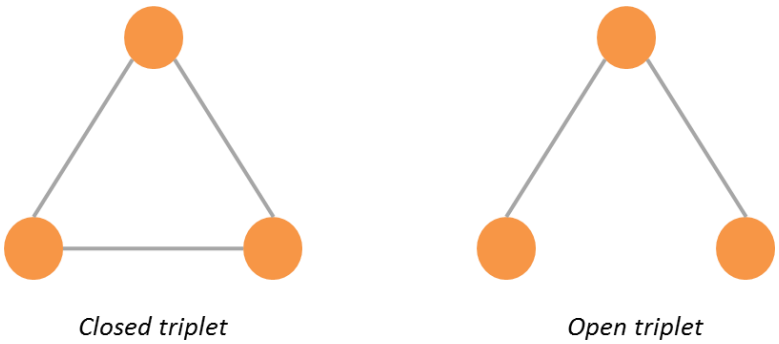


Figure 4.7. An illustration of a closed triplet and an open triplet.

Furthermore, a sample of 'uncertain' predictions are visualised in html enabling the manual assessment of approximate F-measure (Precision and Recall) and to generate some comfort and a feel for performance.

The ER solution is designed to generate a number of features across as wider net as possible, attempting to capture all relevant pairs – hence achieving a good Recall. And then use this set of metrics to determine which predictions are valid and which invalid using a variety of approaches. All of the elements within the model are tied together in a coherent conceptual model. These aspects ensure the model is generic and responsive enough to deal with a heterogeneous set of data types at

the coarse grained level. The parameter Tolerance is provided for fine grained tuning if necessary, although testing indicates only a very small difference.

Problem 9: Different users may have a completely differing view dependent on their domain context, and therefore want to apply the ER in differing ways. For example, within the counter-terrorism domain ER predictions of persons of interest may want to be applied liberally, with every prediction manually validated, whereas in a complex tax crime context involving millions of ER predictions a very conservative approach may want to be taken. Additionally, users may want to simply adopt the binary classification provided or more advanced users may want to use a broader set of metrics and conduct a second phase modelling exercise to create adjunct metadata on the predictions – such as in-situ prediction. The table of predictions provides a binary classification [0,1], an uncertainty rules based model [0-1], a machine learning (either SVM or recursive partitioning) probability classification model [0-1], in addition to a range of granular metrics. Furthermore, the machine learning probability classification model generates model performance evaluation metrics including Kappa coefficient, F-measure, and Logloss. These metrics in combination give the end user a clear reading on the machine learning performance from the perspective of imbalanced classes taking chance into account (Kappa), ignoring chance and class imbalance focusing on Precision and Recall (F-measure), and a metric (Logloss) to measure how close the probabilities are to a perfect distribution of perfectly scored observations, with scores closer to 0 indicating higher accuracy.

Providing a range of metadata at the model level and at a prediction level enables the user to make contextual informed decisions in how to deploy the predictions. For example, a user may see that the ER model metrics (global transitivity and mean diameter) and machine learning metrics (Kappa, F-measure, and Logloss) are very good and so has more confidence in using the machine learning probabilities and applying a lower threshold. Generating a range of metrics is a significant improvement over rules based ER solutions that provide coarse rankings based on the rules used.

Problem 10: When employing vertex contraction how can we decide what attributes to retain and how do we do this at speed. Vertex contraction, otherwise referred to as merging, is the process of representing multiple original entities – that represent the same real-world entity – as a single entity in an alternate resolved or merged representation of resolved entities. Reducing multiple entities into a single entity has repercussions for how to efficiently store and retrieve attributes about each newly contracted entity. There are many solutions to this, including selecting the entire set of attributes from a single primary entity. Taking this approach has the benefits of speed and simplicity. Methods that measure the quantity and quality of each entities set of attributes, including the provenance of the data source can be used to determine which entities attributes to favour. Alternatively, semantic approaches can be used to represent all of the attributes and use an ontological approach to enable the

user to determine which attributes to materialise. The solution applied here takes each set of entities to be contracted and goes attribute by attribute making a contextual decision on which was the best attribute to retain. These primary attributes are selected based on simple rules such as string length and frequency (e.g. the string "Raj Singh-Smith" is longer than "Raj Singh" and so will be retained). The original graph is retained in memory and each entity's id is traceable enabling a user to toggle between the newly contracted entities (and their attributes as presented) and the original entities (and attributes) that constitute the newly created entity. The solution implemented is based on a table-based approach that uses modification in place, generating a new graph object rather than altering an input graph. Using this approach minimises computational expense, both in terms of speed and memory.

Problem 11: ER solutions often have to be manually 'tuned' through user experimentation to find the optimal ER settings. How can ER be generalizable enough to deal with heterogeneous datasets? Every solution has its limitations. The limitation of this solution is that it is computationally expensive, both in terms of runtime and RAM usage, when compared to market competitors. Therefore, the ceiling of scalability is generally limited to the tens of millions. Part of the reason for this is that the model focuses on the detection of complex non-obvious pairs of entities, the last few % of accuracy which is extremely challenging. Therefore, as the solution has been developed with the most complex ER problems in mind datasets with 'simpler' resolution problems are within scope – the inverse is not true.

The solution determines how to process the data based on parameter inputs (e.g. the Tolerance parameter) and properties of the data. An example of how properties of the data influence the model is the PNOC. This classifier generates features directly from the data in a typical feature engineering process. Features such as number of characters in each proper name, the number of vowels in each proper name, etc. but also generates 'indirect' features such as name assortativity. Name assortativity is based on how likely a person's name is 'associated' to another person from the same classification versus someone from a different classification.

Secondary sources of data provide specific context to support data driven models. The input data will rarely provide all of the context that is required to make good ER decisions. For example, the existence of entities with proper names that have a Slavic origin may provide difficulties for models that are tuned to anglo proper names. How can secondary sources of data support 'Aleksandr' == 'Sasha' and 'Aleksandr' == 'Alex', but 'Alex' will not always be synonymous with 'Sasha'. The solution used is firstly simply using a list of proper names to support the proper name classification model and secondly representing the onomastic gazetteer as a directed graph, enabling very specific onomastic relationships to be extracted and used in computationally efficient ways.

In addition to the novel framework of the ER solution, numerous elements that comprise the ER solution are novel and non-trivial enough to require further explanation. In the following section we will detail the following features:

- Proper Name Classifier
- Proper Name Origin Classifier
- Reference Graph Algorithm
- Collective ER (Contextually applied Transitivity (transitive closure) and Contextual Exclusivity)
- In situ ER prediction

4.1.7 Novel computational solutions – a detailed view

4.1.7.1 Proper Name Classifier (PNC)

Problem

ASM metrics measure the edit distance between strings. ASM metrics are commonly used in ER to detect misspellings between pairs of names which are actually equivalent (e.g. “Carl” and “Carrl”). However, while this is a successful strategy, a small number of pairs have a low edit distance but rather than containing a misspelling are just similar proper names (for example, “Joan” & “Jason”; “Farzad” and “Farad”; “Juan” and “Jun”, and “Carl” and “Carla”). A secondary problem, subsequent to the detection of pairs containing proper names that are similar, is how do we then distinguish between whether the proper name pair contain a proper name by mistake rather than this pair actually representing two different real-world entities. How do we discriminate between these two classes?

Goal

The goal of the PNC is twofold. Feed inputs of each pair’s family and given names and determine whether any of the pairs name combinations both contain proper names or names that are more likely derived from error (e.g. transcription, transliteration). Then for every pair that does not contain a name with an error and has a ‘low’ edit distance, determine the probability of whether the pair refers to two distinct real-world entities. The output is the identification of a set of predictions that are invalidated as false positives.

Purpose

This classifier is used as a filter to identify instances where ASM metrics have falsely determined a pair's proper names are equivalent. For example, the names "Carl" and "Carla" are proper names that are very similar and therefore most ASM's will score names containing this pair as low distance / high equivalence. This pair is unlikely to be the same real-world entity so in this case the prediction derived from the ASM score is retained but classified as 'invalidated' using the PNC.

Design

The problem of distinguishing between proper and non-proper names is complex and the frequency distribution is marked by a long tail. This means that as attempts are made to increase model performance, pushing into the tail of the distribution (i.e. those names that are less common), error increases and at some point the efficacy of the model diminishes. Therefore, a pragmatic modelling decision was made to build a highly performant classifier that is limited in terms of its ability to identify proper names from non-proper names (i.e. names derived from typographical error). Gazetteers are the favoured approach to identify proper names within named entity recognition research (Nadeau & Sekine, 2007) and we will be using a gazetteer in combination with utilising the target data itself. However, NER itself does not generally focus on whether the proper name detected contains a typographic error, nor whether the proper name detected was written in error.

The design of this classifier is such that it targets a very specific entity resolution sub-problem – discriminating between atomic proper names and non-proper names generated through unintentional typographic error or intentional name manipulation, and then makes a secondary classification on whether either of the proper names within a pair are derived from error based on frequency of the proper names.

As such, the size of the set of predictions that the classifier influences is small and varies in line with the extent that the target data contains proper names that are very similar. Some cultural practices and the set of proper names generated from these practices will result in proportionally larger sets of similar names. This may be driven through the presence of a smaller set of popular similar names (e.g. "Arusha", "Anusha", "Anushri", "Anish", "Anisha") in conjunction with common family names (e.g. "Singh", "Patel") and given name convention where siblings share the same middle name, in the context of a large dataset so probabilistically there is a higher likelihood of people having similar names and dates of birth. Additionally, specific naming convention such as Arabic that contains a limited set of specific patronymic (nasab) and paedonymic (kunya) names. Patronymic refers to a name derived from a male ancestor (e.g. "son of" or "ibn" or "bin" or "bint") and paedonymic refers

to a name derived from one's child usually applied metaphorically (e.g. "Abu Bakr" translated as "father of a young camel").

The classifier firstly extracts all atomic proper names identified in more than *n* unique full names from the target data into a vector, and transforms this vector into combinations of pairs. The determination of *n* (default of 20) is based on experimental experience and future versions can optimise how to identify a more optimised set of proper names.

A subset is then drawn from this set of pairs by accepting any pair that generates a Levenshtein edit distance of between 2 and 6, AND a Jaro-Winkler edit distance of between 0 and 0.6. These thresholds were experimentally derived as a useful heuristic for determining the cut-off for which relevant pairs to retain, but of course optimisation is a future step. Creating a subset of pairs in this way ignores those pairs that are exact, almost exact and not similar at all. The reason to ignore almost exact matches at this stage was that these are predominantly proper name variants rather than differing proper names (e.g. "David" versus "Davide"). Failure to include sub-setting would result in both inefficient run-time in large datasets, and increased error.

Next a gazetteer of pairs of curated proper names, sourced from a range of global regions, that are similar but actually different is added to all pairs of proper names commonly found in the data. This results in a two column data object containing a range of proper name pairs that have a minimal 'edit' distance. The onomastic gazetteer is then used to ensure no pairs with onomastically equivalent names are included (e.g. we want to ensure we keep pairs like "Kev" == "Kevin"). This object – let's call it the 'proper name graph' – is used as a lookup graph to identify instances of when a pair predicted to be a match merely reflects similar, but not equivalent, proper atomic names.

So, the input set of prediction pair names are compared to the proper name graph using a tuple based approach aiming to identify any pair that contains two different proper names (e.g. "Roland" and "Ronald"), ensuring that compound names and name transposition are identified and treated appropriately. For example, the two tuples {"Chan", "Hai", "Ming"} and {"Chan", "HaiMing"} can generate the pairs {"Chan", "Chan"}, {"Hai", "HaiMing"}, and {"Ming", "HaiMing"} which could lead to the determination that both "Hai" and "HaiMing" and "Ming" and "HaiMing" are pairs of different proper names. Resulting in the invalidation of the prediction that "Chan Hai Ming" and "Chan HaiMing" are the same real-world entity. The tuple based approach used ensures that compound names and name transposition are identified and removed from the set of predictions that are invalidated as false positives.

We have previously generated a proper name graph from which we can identify predictions that contain proper names that are similar but different (e.g. "John Brian Higginbotham" and "John Brain

Higginbotham"). The second problem to solve is the subset of pairs that contain differing proper names due to a typographic error – often generated from transposition error (e.g. "Brian" versus "Brain"). The solution to this problem is to determine the frequency of each proper name within the data in comparison to how often each proper name is represented in a prediction pair that contains two different proper names. For example, within the NZCO data there are 16,966 instances of "Brian" and 132 instances of "Brain". Within the subset of prediction pairs that contain different proper names "Brian" and "Brain" is represented 118 and 120 times respectively. In other words "Brian" and "Brain" have been identified in 118 and 120 pairs of potential false positives. This frequency in the context of the frequency of how often the proper name is noted within the target data is sufficient to heuristically classify prediction pairs containing "Brian" and "Brain" as derived from error.

In this way we can determine the estimated proportion of times a name has been used in error, and as such how likely the existence of that name in a pair is due to error rather than simply two different real-world entities. For example, "Brian" has a proportion of 0.007 and "Brain" has a proportion of 0.909. All pairs that contain a proper name that has a proportion exceeding 0.5 (that is the proper name is represented more than half the time as a potential error) are heuristically classified as derived from error. Again this cut-off of 0.5 is derived from experimentation, however future work should determine an approach to optimise this heuristic method.

The output from the PNC is an integer vector of those pairwise predictions that contain proper names that are similar but different (and not generated from error) and hence can be considered a false positive. This output is then used to update the prediction metadata, retaining provenance.

PNC performance

The performance of the PNC is based on the experimental results from testing the classifier on the Sanctions, Dark Network/STR, Offshore Leaks and NZ Companies Office data (see table 4.3).

Table 4.3. Proper Name Classifier (PNC) evaluation results.

Evaluation of the Proper Name Classifier

	Sanctions	Dark Network / STR	Offshore Leaks	NZ Companies Office
Data				
Vertices	~23,000	~360,000	~1.4 m	~16 m
Edges	~44,000	~900,000	~2.4m	~90 m
Persons	~14,000	~100,000	~400,000	~7 m
Organisations	~7,000	~15,000	~640,000	~4 m
Scalability				
# ER predictions	40,854	190,711	584	2,223,885
# PNC predictions	91	14	6	911
Runtime (seconds)	19.79	73.55	23.02	1,356.22
Accuracy				
True Positives ratio	0.92	1	1	*0.94
ER model performance (No PNC | PNC)				
Global transitivity	0.7475 | **0.7487**	0.9999 | 0.9999	1 | 1	0.9998 | **0.9999**
Diameter metric	0.9026 | **0.9054**	0.9989 | **0.9990**	1 | 1	**0.9851** | 0.9752
Precision*	0.9524 | **0.9804**	**0.9901** | 0.9804	0.9348 | **0.9792**	**0.9992** | 0.9984
Recall*	0.8621 | **0.9804**	0.9901 | 0.9901	0.9773 | **0.9792**	0.9894 | **0.9907**
F measure*	0.9050 | **0.9804**	**0.9901** | 0.9852	0.9556 | **0.9792**	0.9943 | **0.9945**
Persons contracted	5,904 | **5,911**	56,303 | **56,316**	519 | **522**	**4,168,483** | 4,127,790

* Sampling used to estimate accuracy

The key metrics to determine success is weighing the improvement in performance with the cost of runtime, within the context of the already high performance achieved. We can see that the number of classifications made by the PNC is relatively small in relation to the overall set of ER predictions. The PNC classifications being those ER predictions that include non-onomastically related proper names, or, in other words, includes those pairwise predictions that contain conflicting proper names – excluding nicknames.

The PNC is used three times within the ER model to focus on specific sub-problems. The results here represent the PNC being executed on all ER predictions derived from the Non-Obvious sub-module. Subsequent executions within the ER model are conducted on very small subsets of data, with relatively trivial runtimes. In all, the PNC consumes between 2-5% of the ER model runtime. Scalability, beyond what has been experimentally tested here, is a further consideration that requires additional testing. Given the sizes of the datasets under evaluation scalability was not an immediate issue however there is potential to deploy this classifier on big data within a distributed computing platform as the data is table based and code is executed in a serial way - therefore making the engineering of this classifier into a parallel or distributed computing context a relatively simple exercise.

The size of the number of predictions detected by the PNC as false positives across the four datasets is very small. However, the highly accurate PNC generates better quality metadata which has a cascade effect throughout the ER model, manifested most explicitly in the measurement of uncertainty. The

sub-modules that consume the higher quality metadata naturally perform better generating higher quality ER predictions, beyond what would be expected. The evidence for this is that if the PNC is designed to detect false positives, so the number of persons contracted should decline in combination with an increase in accuracy metrics (global transitivity, diameter, Precision and F-measure). Specifically, Recall should be unaffected as its calculation is independent of false positives. However, we don't observe this. The number of persons contracted actually increases in combination with accuracy metrics across the Sanctions, Dark Network and Offshore Leaks data, but within the NZCO data we do observe the number of persons contracted decreasing with only an observed increase in global transitivity. These observations, of course, need to be taken in the context of the variability of Precision, Recall and F-measure, as these metrics are based on samples and the human validation is a subjective exercise. The levels of accuracy in the ER model are such that actual performance enhancements are difficult to detect due to the high baseline performance. For example, the Precision calculation of 0.9984 for NZCO using the PNC involved one false positive within 636 observed predictions, so the difference between 0.9984 and the non-PNC Precision score of 0.9992 merely reflects a larger sample. Furthermore, global transitivity and diameter are useful markers of estimated accuracy but are highly influenced by the nature of the target data.

By nature of the target data we are referring to elements such as the base level of curation and typographic error of each dataset in combination with the volume of entities. The Sanctions and Dark Network/STR data are relatively well curated, however Offshore Leaks and NZCO data are not. The Offshore Leaks data has been extracted from documents via manual and OCR (Optical Character Recognition) mechanisms, which are both subject to high typographic error.

The NZCO data has limited data validation so the quality of data entry is not high, with similar typographic error proliferating through the data (e.g. "Christopher" vs "Chirstopher") and thus common typographic errors can be recognised as proper names due to their frequency ("Chirstopher" has a unique frequency of 24 as a first given name in the NZCO data). Another source of error in the classifier is failing to identify the onomastic relationship between the proper names (e.g. "James" vs "Jimmy"), which is simply due to the onomastic gazetteer failing. The solution is simple and involves annotating the onomastic gazetteer with more onomastic pairs of proper names. Determining proper names as derived from error is a complex endeavour. The attempt here is to mimic a human's decision-making process, yet doing so in a computationally efficient way, and leaving headroom for uncertainty so the model retains high accuracy. Future versions of this code can extend this classifier to the point where it identifies a larger set of false positives that contain similar but different proper names and retains accuracy.

Part of the reason for overall performance improvement is that the existence of the false positive ER predictions creates downstream error in the Collective ER sub-module. So, the cost of falsely attributing correct ER predictions as false positives is outweighed by the benefit of correctly identifying the false positives. The amount of significance is largely determined by the nature of each differing dataset, with datasets that contain fewer typographic errors likely to gain less impressive performance gains.

Notwithstanding these specific "blind spots" the classifier performs well enough to improve the performance of the overall ER model across the four datasets, in terms of increasing accurate predictions, both in terms of reducing false positives directly and false negatives indirectly due to the enhanced quality of metadata.

To summarise, the performance of the PNC is highly specific and accurate, and yields a small but significant improvement to the overall ER model.

4.1.7.2 Proper Name Origin Classifier (PNOC)

Problem

Firstly, from a data perspective the problem is that person proper names, in terms of their linguistic features, are not homogeneous. So, applying blunt metrics that measure differing elements of distance/similarity across a heterogeneous set of names is going to result in measurements that are variable and lack context for consistent accurate interpretation.

Secondly, from an applied computing perspective pairwise comparison is computationally costly, in terms of runtime, RAM expenditure, and generating noise in features by comparing pairs so broadly that through chance metrics indicate equivalence in pairs that are not the same real-world entity.

Thirdly, from a very specific ER sub-problem perspective, people can sometimes use multiple names. A common source of multiple given name use is in instances where a person has adopted a given name from the country where she or he is domiciled to help assimilate into the local culture. From an English speaking perspective this is known as anglicisation. Any metadata giving insight into the origin of a person's given and family name will create visibility over whether anglicisation has potentially taken place.

Fourthly, from a data cleansing perspective we can use knowledge of name origin to infer or target subsets to harmonise data for performance uplift. For example, the Arabic nasab (patronym) can be harmonised so the variants of ibn/bin and bint/bte can be treated synonymously.

Fifthly, the probability of having an association to another person with the same name origin – assortativity – is another application of the PNOC. Using the PNOC in link prediction is not covered in any detail within this section.

Goal

Use the concept of proper name origin as the basis to build a classifier that identifies the origin of the given name(s) and family name(s) of each person. This latent knowledge can be applied in various ways throughout the model using this context to improve decision-making, in terms of what pairs to generate metrics on, and how to interpret generated metrics.

Specifically, this metadata can be used to address the second problem via blocking. Blocking is an approach that helps reduce the computational cost, but is often at the cost of reduced accuracy. Meta-blocking is the application of multiple blocking strategies in combination to reduce sub-sets of pairwise comparison even more. The goal here is to create more accurate targeted sub-sets, or blocks, of entities to compare, increasing speed, reducing RAM, and increasing accuracy.

In response to the third problem identified, people's use of anglicised variant names, requires metrics that enable the identification of instances where persons names contain both the local name origin and a non-local origin. The goal here is to identify entities that use anglicised name variants, and utilise this latent knowledge to make better decisions.

Purpose

The PNOC has been built to predict and classify the origin of each person entities name, and through this enabling a more precise application of ER model sub-modules and thus deliver an enhanced ER performance, both in terms of accuracy and potentially run-time.

Design

The origin derived heterogeneity of person's proper names is intuitive however formally quantifying this heterogeneity is useful. This quantification has been demonstrated through analysis of the person entities extracted from the four evaluation datasets (see table 4.4).

Table 4.4. Proper Name Origin Classifier (PNOC) meta-blocking linguistic features for the four evaluation datasets.

	Median number of characters per name	Mean number of words per name	Mean number of characters per word	Proportion of proper names
Chinese name origin	10	2.54	4.20	0.09
English name origin	16	2.67	6.02	0.75
Latin name origin	17	2.86	6.31	0.04
Russian name origin	15	2.37	6.73	0.01
English/Chinese name origin	10	2.42	4.68	0.03
Arabic name origin	14	2.60	5.98	0.02
Persian name origin	17	2.52	7.13	<0.01
Subcontinent name origin	13	2.42	6.02	0.01
NA	-	-	-	0.05

We can see through the simple analysis illustrated in table 4.4 that there is significant variance between the classes across the four metrics used. This analysis focuses on the difference in name size, but of course there are a number of other linguistic elements that contribute to the differences (e.g. syllables, n-gram, use of compound names). For our purposes here the more superficial level of proper name size is sufficient to demonstrate the point. To give a comparative example Arabic and Chinese origin proper names have a maximum of 5 and 3 atomic words, an average of 5.98 and 4.20 characters per atomic word, and a median total of 14 and 10 characters, respectively. Any metric used to quantify the distance or similarity between strings is going to be sensitive to input variance of this kind.

So, the blunt classless approach interprets ASM metrics across all pairs without taking name origin into account. In this way we are using the same approach to interpret what an ASM score means whether it is comparing a pair of Chinese or Arabic names. An edit distance of 2, for example, is quite different in each context. However, when we have knowledge of the origin class the application and interpretation of the metric can be contextual, and therefore boost performance.

The diversity of proper name origin classes does not afford the opportunity for the same Precision that a more homogeneous set would. So, the obvious solution is to incorporate the origin of proper names into the ER model where useful and take this variability into account when predicting whether the pair of person entities refer to the same real-world entity or not. Making use of the latent knowledge of name origin can be applied in many ways. Here are three examples, relating to the problem stated earlier.

Blocking

The PNOC prediction is used as a meta-blocking mechanism, enabling an increase in performance (speed and accuracy) when undertaking pairwise equivalence. Meta-blocking is a class of blocking

algorithm that can be used in conjunction with blocking algorithms to create more granular blocks from which to compare pairs. A common approach is the canopy approach that is used to increase the granularity or generate overlapping blocking sets to make the ER either, more scalable and quicker, or more accurate (McCallum, Nigam & Ungar, 2000).

Traditional ER blocking tends to identify subsets (block) of entities based on a superficial level (i.e. the characters and their order within a person's name). However, there is much more latent knowledge – at both an atomic and aggregate level – available to enable a significantly better matching result. Let's look at a fictitious example to highlight what we mean.

For example,

"TSEON SHIH FENG"	Truncation == 'F'	Soundex == F520	FNRG == 358	PNOC== 'CN'
"FENG SHIH TSEON"	Truncation == 'T'	Soundex == T250	FNRG == 358	PNOC == 'CN'
"WERNER MOELLER TYSON"	Truncation == 'T'	Soundex == T250	FNRG == 88	PNOC == 'Other'

Truncation and Soundex algorithms place "TSEON SHIH FENG" and "FENG SHIH TSEON" in separate blocks and therefore not subject to comparison. The Family Name Reference Graph (FNRG) algorithm and PNOC has both "TSEON SHIH FENG" and "FENG SHIH TSEON" in the same Chinese origin meta-group. However, "FENG SHIH TSEON" and "WERNER MOELLER TYSON" have been placed in the same Soundex derived block ("T250"), but different FNRG and PNOC blocks, even though the names are clearly not similar and indeed clearly have differing origins.

The above illustrates how not only can the use of the PNOC improve computational efficiency by ensuring like are compared to like, but also enables the design of much more precise functions to determine similarity based on the understanding of each groups aggregated linguistic features. In other words examining the linguistic features of proper name origin classes at the aggregate level generates clear heterogeneous semantic profiles of each class. So, a more nuanced approach that takes this semantic context into consideration is going to perform better, especially in large datasets where the large number of operations generates an increasingly high likelihood of measuring high pairwise equivalence where none exists.

Anglicisation

The PNOC predictions are also used when attempting to find pairs that contain an anglicised proper name (e.g. ENCN). The specific class of the proper name can be used, however the accuracy of the classifier is critical to ensure that its use does not generate more error than it is worth.

Harmonisation

Finally, PNOC predictions are also used to target name sets for specific harmonisation treatment to ensure we create the best chance of finding true positives.

Classifier design

Previous models focusing on this problem include n-gram statistical methods (Nobesawa & Tahara, 2005), a Maximum Entropy classification utilising an ontology including relationships between origin grammar and linguistic features – including n-grams – (Fu, Xu & Uszkoreit, 2010), and Bhargava and Kondrak (2010) used n-grams and word length as features for a SVM. All three of these approaches showed encouraging results. The statistical n-gram approach of Nobesawa & Tahara (2005) attempted to identify the origin of proper names from 12 countries with accuracy ranging from 50 to 93%, the Maximum Entropy based classifier, utilising n-grams and linguistic features, attempted to identify the origin of proper names from 8 countries with accuracy ranging from 73 to 98%, and the SVM approach, using n-gram and word length, generated accuracy results ranging from 94 to 99% across four classes. Unfortunately, neither runtime nor scalability was explicitly mentioned in these papers, crucial elements to applied computational models. Initial testing was therefore built around testing the viability of n-gram based models in terms of accuracy, runtime, and scalability. It was quickly established that the generation and utilisation of n-gram metrics produced an unacceptably high runtime cost (e.g. ~10 times slower) with no accuracy benefits in comparison to the PNOC and so alternative features were explored whilst retaining the machine learning framework.

The design of the PNOC is a machine learning approach that uses engineered features as inputs into either a recursive partitioning (RPART) or a support vector machine (SVM) algorithm to make a binary or multi-classification on the origin of each person's name. Classifications can include compound classifications where a proper name has multiple origins. A feature of the classifier is that it can focus on the full range of origin classification possibilities or reduce the complexity of the model to work as a binary classifier by focusing on a single origin. The currently implemented set of origins include; Arabic, Chinese, English, Japanese, Persian, Slavic, Subcontinent, Spanish, and West African. Of course, this set of name origins do not cover the full set of possibilities, and is slightly different concept than the country concept used by Nobesawa and Tahara (2005) and Fu, Xu and Uszkoreit (2010).

The engineered features include:

1. number of characters per name
2. number of words per name
3. mean number of characters per word
4. number characters in given name
5. number of characters in family name
6. number vowels per word
7. number vowels in given name
8. number vowels in family name
9. Name origin reference graph community proportion - Family name [Binary only]
10. Name origin reference graph community proportion - Given name [Binary only]
11. Name origin reference graph community ordinal - Family name [Multi only]
12. Name origin reference graph community ordinal - Given name [Multi only]
13. Count of the use of letters from A to Z in each given and family name

The majority of the engineered features are self-explanatory; however the Name Origin Reference Graph requires some explanation. Using the same technological approach as the Reference Graph Algorithm a graph is generated by using each unique atomic proper name as a vertex and drawing edges between vertices when there is a co-occurrence of names. Each word that constructs a person's proper name represents a separate node, and the co-occurrence of each pair of words that constructs a person's proper name is used to create a weighted edge.

For example, the following names {"DANIEL ANDRES COSTA GONZALEZ", "DANIEL ALEJANDRO GORRIN GONZALEZ", "DANIEL SMITH", "DANIEL SMITH", "CHEN SHIH FONG", "CHEN SHIH FENG"} would generate the graph in figure 4.8 (see below). The resultant graph, dependent on the size and heterogeneity of the set of names derived from the dataset, may be sparse and contain multiple components. This sparsity is reduced by applying an ASM algorithm to incrementally add enough edges between words to optimise the number of communities generated by the community detection algorithm - Louvain method (Blondel et al., 2008).

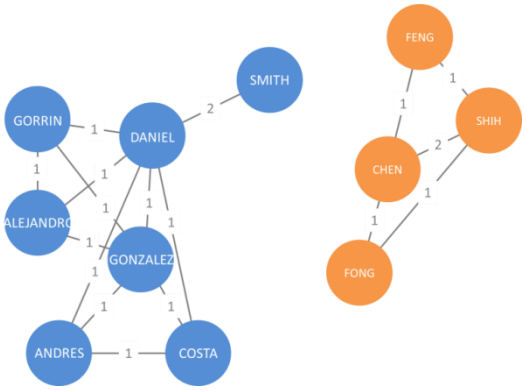


Figure 4.8. This figure illustrates how the Name Origin Reference Graph is constructed.

Then this weighted graph is partitioned into communities. Each community of atomic names tends to cluster names of the same origin together. The binary classifier option then determines the proportion of a single origin class – for example the proportion of Chinese names in a community using secondary sources of data that contain a range of proper names with an associated origin. Let's call this secondary source of data the name origin gazetteer. The multi classifier uses a technique that identifies the most frequent origin class and then numerically ranks the communities on this basis, with the goal of transforming the categorical notion of communities into an ordinal scale of sorts with clusters of similar communities located proximally on the scale.

A 'ground truth' set of data is generated automatically from the name origin gazetteer which is used as a lookup to identify the 'ground truth' set.

Within the model building phase a random training set with known classes is used to generate a model to then apply, via parallel processing, to the full set of data where classes are largely unknown. The size of the training set is limited to 80% of known cases, to a maximum of 15,000 cases. Significant experimentation was conducted on the SVM model across both the Dark Network/STR data and NZ Companies Office to tune the parameters for generalised performance.

The output of the classifier includes a multi-classification vector, including compound classes, when the multi-classifier parameter is selected, or a binary classification vector (plus compound class), when the binary classifier parameter is selected. Accuracy and Cohens Kappa coefficient is then generated from the model's ability to correctly predict the known classes.

Performance

The performance of the PNOC is based on the experimental results from testing the classifier on the four evaluation datasets, as these datasets best cover the spectrum of name heterogeneity. Table 4.5 shows the test results over the range of evaluation data sets, by runtime, accuracy, and Cohen's kappa coefficient. Accuracy is determined by assessing all of the predictions for the cases which are known (ground truth) and simply performing the following calculation; 1 - (wrong / correct). Cohen's Kappa coefficient measures inter-rater agreement taking into account the probability of agreement occurring by chance, and thereby taking into account imbalanced classes.

Table 4.5. Illustrates the performance of RPART and SVM algorithms on differing sizes of name sets extracted from the four evaluation datasets.

Performance Measures

Sanctions

			R P A R T			S V M	
	Number of names	Run Time	Accuracy	Kappa	Run Time	Accuracy	Kappa
Binary	8,000	11.38 s	0.9710145	0.8802083	25.73 s	1	1
	14,000	15.20 s	0.9864407	0.9554061	28.47 s	0.9966102	0.9892438
Multi	8,000	11.12 s	0.8702290	0.8385660	17.72 s	0.9809160	0.9764288
	14,000	12.07 s	0.9160689	0.8962216	21.86 s	0.9885222	0.9859601

Dark Network/STR

			R P A R T			S V M	
	Number of names	Run Time	Accuracy	Kappa	Run Time	Accuracy	Kappa
Binary	8,000	6.70 s	0.9831461	0.8812317	18.81 s	0.9937578	0.9558776
	16,000	8.44 s	0.9890282	0.9173497	22.98 s	0.9949060	0.9621676
	32,000	9.19 s	0.9852126	0.9513503	21.95 s	0.9963031	0.9877527
	64,000	11.36 s	0.9774720	0.9581991	27.80 s	0.9985729	0.9973251
	110,000	15.81 s	0.9820910	0.9663222	40.40 s	0.9988267	0.9978011
Multi	8,000	9.15 s	0.9653145	0.8528849	22.67 s	0.9923574	0.9683033
	16,000	9.38 s	0.9605696	0.8403123	37.58 s	0.9930632	0.9725770
	32,000	11.44 s	0.9387495	0.8420719	47.78 s	0.9929817	0.9832125
	64,000	12.82 s	0.9417297	0.9004819	90.16 s	0.9955400	0.9924654
	110,000	19.61 s	0.9351205	0.8872820	136.89 s	0.9969496	0.9948366

Offshore Leaks

			R P A R T			S V M	
	Number of names	Run Time	Accuracy	Kappa	Run Time	Accuracy	Kappa
Binary	8,000	11.12 s	0.9837869	0.9213786	23.93 s	0.9965258	0.9830732
	16,000	13.22 s	0.9878184	0.9320913	44.96 s	0.9964009	0.9798062
	32,000	17.87 s	0.9845501	0.9215283	33.49 s	0.9962133	0.9804424
	64,000	34.15 s	0.9807295	0.9146472	57.45 s	0.9957846	0.9807827
	128,000	77.23 s	0.9890735	0.9318801	88.56 s	0.9968846	0.9807040
	256,000	167.35 s	0.9801437	0.9462104	162.67 s	0.9926256	0.9796795
	300,000	177.66 s	0.9735551	0.9352526	203.00 s	0.9906001	0.9768662
Multi	8,000	12.20 s	0.9565217	0.8599235	20.28 s	0.9876543	0.9616954
	16,000	10.06 s	0.9610794	0.8636674	55.80 s	0.9924754	0.9739353
	32,000	17.19 s	0.9555429	0.8419489	48.16 s	0.9917355	0.9721114
	64,000	35.63 s	0.9499962	0.8684796	137.07 s	0.9906435	0.9755784
	128,000	62.00 s	0.9586731	0.8813989	242.87 s	0.9875900	0.9643562
	256,000	126.08 s	0.9185295	0.8455899	418.86 s	0.9788104	0.9615942
	300,000	195.01 s	0.9021763	0.8337554	409.20 s	0.9779366	0.9624737

NZ Companies Office

			R P A R T			S V M	
	Number of names	Run Time	Accuracy	Kappa	Run Time	Accuracy	Kappa
Binary	8,000	22.74s	0.9778565	0.6127833	16.07 s	0.9964570	0.9430962
	16,000	9.12s	0.9877607	0.8011319	27.68 s	0.9977335	0.9660296
	32,000	11.42s	0.9888674	0.8252740	20.40 s	0.9975799	0.9639822
	64,000	24.86s	0.9887441	0.8319883	38.50 s	0.9979858	0.9726941
	128,000	24.27 s	0.9917836	0.8728997	41.59 s	0.9991004	0.9860283
	256,000	51.92 s	0.9919791	0.8803936	91.86 s	0.9973864	0.9611144
	512,000	217.92 s	0.9900921	0.8746131	307.48 s	0.9967629	0.9589151
	1,024,000	460.01 s	0.9890136	0.8958000	527.30 s	0.9956086	0.9586701
	2,028,000	953.95 s	0.9886607	0.8937123	764.32 s	0.9954777	0.9571058
	4,056,000	1,965.06 s	0.9876831	0.8922507	1,797.72 s	0.9948821	0.9546362
Multi	8,000	14.62 s	0.9725664	0.3470643	24.34 s	0.9973451	0.9590010
	16,000	28.66 s	0.9738031	0.5830996	32.88 s	0.9986450	0.9834105
	32,000	15.44 s	0.9778580	0.7349913	28.35 s	0.9954272	0.9491345
	64,000	33.55 s	0.9790588	0.7515366	60.57 s	0.9965882	0.9636650
	128,000	32.61 s	0.9815432	0.7644412	110.71 s	0.9971422	0.9663601
	256,000	67.67 s	0.9823766	0.7811867	267.95 s	0.9946325	0.9370012
	512,000	245.12 s	0.9786197	0.7661889	416.13 s	0.9922731	0.9212285
	1,024,000	526.76 s	0.9750628	0.7905073	629.08 s	0.9893533	0.9150384
	2,028,000	1,048.11 s	0.9746498	0.7894028	1,358.65 s	0.9888074	0.9117464
	4,056,000	2,338.28 s	0.9754149	0.8188043	3,105.10 s	0.9888471	0.9198190

The accuracy levels of both RPART and SVM approaches across the four datasets range from 97-98% and 99-100% for the binary classifier respectively, and 87-98% and 97-99% for the multi classifier. This indicates the PNOC outperforms the n-gram based approaches outlined above, however it is important to note the differences in that the PNOC was targeting classes at a more natural linguistic origin level rather than the more artificial boundaries of country - the goal of determining whether a proper name belongs to that of someone from the US or UK is extremely ambitious. We have no doubt that to achieve consistent accuracy more data beyond atomic proper name is required to delineate between classes on a country basis. Comparing RPART and SVM we can observe a significant uplift in accuracy when using SVM, however there is an associated runtime cost. However, this runtime cost interestingly diminishes as the size of the data increases, to the point where the SVM starts to outperform the RPART on the binary classifier around the 2 million vector length mark.

In terms of scalability the runtimes show an approximate linear runtime performance, at least to the scale of ~2 million unique names.

Taking the broader perspective of performance in terms of how well the overall ER model performs when we compare utilising RPART versus SVM in the PNOC we see interesting results, as per table 4.6.

Table 4.6. Illustrates the contribution of RPART and SVM algorithm to the overall ER models performance across the four evaluation datasets.

Performance of the PNOC: SVM v RPART

	Sanctions **SVM | RPART**	**Dark Network / STR** **SVM | RPART**	**Offshore Leaks** **SVM | RPART**	**NZ Companies Office** **SVM | RPART**
Runtime (seconds)	518 | **457**	1,607 | **1,500**	3,223 | **3,083**	**55,507** | 57,620
ER model performance				
Global transitivity	0.7471 | **0.7487**	0.9999 | 0.9999	0.9999 | 0.9999	0.9999 | 0.9999
Diameter metric	**0.9119** | 0.9054	**0.9991** | 0.9990	**0.9965** | 0.9950	**0.9778** | 0.9752
Entities contracted	8,281 | **8,321**	**90,237** | 88,831	**70,150** | 70,063	**7,367,445** | 7,366,952

Here we see that runtime is better for RPART for smaller datasets and better for SVM on the largest dataset. Furthermore, we generally see a small consistent improvement on performance metrics when using the SVM version, excluding the Sanctions data. Comparing the ER model with and without the use of PNOC was not carried out, due to the fact that the intrinsic design of the ER model is now so reliant on PNOC that this was simply not feasible.

In terms of impact on the ER model the PNOC is utilised as a core meta-blocking algorithm and as a feature to detect anglicisation. As such the performance of the ER model is significantly reliant on the high performance of this model, both in scalability and accuracy.

In terms of extensions it is clear that over-lapping community detection, or other clustering approach utilising more linguistic attribute data may provide superior results, however at this stage the simple approach outlined above has yielded very good results and as the product matures this element may be enhanced. Also, in terms of engineering the model the code could be rewritten to take advantage of a distributed computing environment.

4.1.7.3 Reference Graph Algorithm (RGA)

Problem

Entity resolution is fundamentally a pairwise problem. As such methods have to be created to navigate the intractability of pairwise computation. Blocking is such a class of methods that is a pragmatic solution to the intractability of pairwise computation. Blocking is a strategy to break a set of target entities into subsets or blocks, with the intent of conducting pairwise computation on each set in isolation. The goal of blocking is to quickly construct optimal blocks of entities so subsequent pairwise performance is then fast, scalable and accurate.

A number of blocking methods have been devised including phonetic algorithms like Soundex (Odell & Russell, 1918) and Double-Metaphone (Philips, 2000). These phonetic algorithms create keys based on the phonetic sound of the string (e.g. Soundex of "Rodriguez" == "R362"). Another collection of blocking methods, the permutation/truncation approaches, generate keys off a pre-set number of character positions from the string (e.g. 1^{st} 2^{nd} 3^{rd} for "Rodriguez" == "ROD"; 1^{st} 6^{th} 9^{th} for "Rodriguez" == "RGZ").

Meta-blocking is a strategy that takes block classes as an input and generates an optimised blocking class to enhance accuracy and computational speed of the subsequent pairwise comparisons. Hernandez and Stolfo (1995, 1998) and McCallum, Nigam and Ungar (2000) outline the window or canopy approach which creates overlapping classes yielding a reduction in computational expense whilst retaining or improving accuracy.

Experiments conducted on the evaluation data quickly concluded that whilst many blocking algorithms were computationally efficient the accuracy was a limiting factor to the overall accuracy of the ER model. In response to this the Reference Graph Algorithm was developed (Robinson, 2016).

Goal

The goal was to create an algorithm that generates metadata useful for blocking purposes in ER, across a heterogeneous range of domains and data sizes. Performance wise the algorithm must generate more accurate blocks than blocking algorithms currently available, within a tolerable runtime, and in a scalable manner. Additionally, it would be beneficial if the output of such an algorithm could be persisted and curated over time so its performance increases and it can be used in a variety of circumstances, such as ER decision-making and even named entity recognition.

Purpose

The purpose of developing such an algorithm is to improve the overall performance of the ER solution, and specifically the ability of the solution to reduce the number of false negatives – missing those pairs of duplicates that represent an entity in the real-world. This high performing ER then creates the opportunity to perform more accurate downstream analytics, such as graph-mining, on the data.

Design

As stated in Robinson (2016) a reference graph, in this instance, is defined as a graph constructed from a set of proper atomic names whose pairwise distance is calculated using a variety of concepts, including string similarity and co-occurrence, and represented as a graph that can be improved over time to enhance ER performance. Improvement to the reference graph can be derived from improvements in the algorithm that constructs the graph, the integration of additional data, or the manual curation by human experts. The RGA generates metadata that is multi-purpose. RGA metadata can be used for blocking and in several other aspects of an entity resolution model. The other aspects include using the RGA as a building block to understand proper name origin and using the output of RGA as a contextual data point when making transitive decisions in collective entity resolution.

Reference graphs are implemented here using two inputs, family names and given names, generating the two outputs Family Name Reference Graph (FNRG) and Given Name Reference Graph (GNRG) respectively.

The RGA generates a reference graph by taking a vector of proper names as an input and measuring the string distance between the vector of unique proper names. The current default algorithm deployed to measure string distance is the Jaro-Winkler algorithm (Winkler, 1990).

If the size of the input vector is sufficiently large to create pairwise intractability an intermediate blocking phase is undertaken to generate a sampled rather than complete pairwise assessment. The intermediate blocking phase is based on conducting a sample of pairwise string distance assessments. This sampling process firstly involves classifying each name based on its first letter (e.g. "Chan" == "C"), and ensuring that a significant sample from each class is then compared to every other class. This exercise generates a complete graph which is then pruned, deleting edges deemed as not relevant. The pruning mechanism used is a threshold. Extensive experimentation has been conducted to determine a reasonable threshold default. The resultant graph creates the basis for determining the strength of relationships between each class. This is achieved by vertex contraction based on class (see Figure 4.9.).

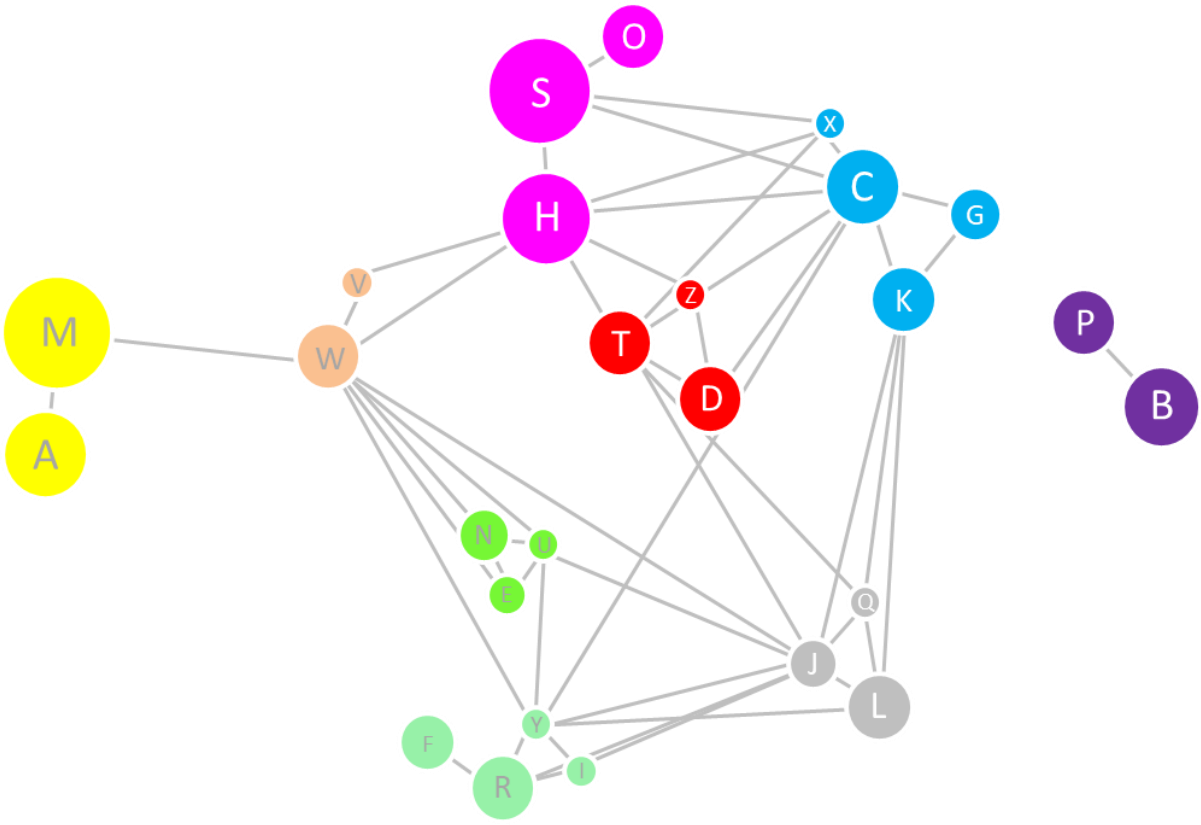


Figure 4.9. This figure illustrates the contracted class based graph showing the relationship between the classes of names based on their first letter.

The second part of the intermediate blocking phase is using the contracted class based graph as the basis to create meta-blocks (as differentiated by the colour of nodes above). This is done by applying a hierarchical based community detection algorithm, the Fast Greedy (Newman, 2004; Clauset et al., 2004) algorithm. This enables creating sufficiently granular communities to form these meta-blocks, using contracted class-based graph metadata and the community detection dendrogram. The meta-

blocks are then used to conduct string distance pairwise comparison between each proper name within each meta-class.

The third part of the intermediate blocking phase is conducting a sample of string distance pairwise comparison between adjacent meta-blocks, and then a final sample across all meta-blocks.

The output from this phase is an approximately complete graph generated from the string distance between proper names. In this implementation the proper names are either Family names or Given names.

Now that we have an approximately complete distance graph of proper names, we have the basis from which to partition the nodes (i.e. proper names) into blocks. The next step is to retain only edges that support the identification of an optimal set of blocks. A simple thresholding approach to prune edges was ruled out as unique names would likely become isolated and result in poorer blocking performance. Instead, the two highest weighted edges incident to every node and edges with a weight over a pre-prescribed threshold (again derived through experimentation) were retained to ensure the graph stays connected and the smallest component is a triad. This creates the optimal conditions for fast and scalable blocking output.

However, before the blocks are generated secondary source data is introduced. The secondary source data comes in the form of names and their relevant name variants. These name variants are a combination of hypocorisms (nicknames) and transliterations. Utilising secondary sources of data in this way creates the opportunity for subject matter experts to add their knowledge and generate instant ER model enhancement. This is an important human centred systems feature which helps create user investment in the technology. This additionally reinforces the idea that data assets, developed and curated by experts, can be incredibly valuable in not just ER but also other technology such as named entity recognition.

The secondary source data is designed to enable the addition of relationships between a name and a name variant, and negate the relationship between two proper names. For example, how the Chinese family name 韩 is transcribed into the Latin alphabet is reliant on the originating dialect. In Hainan 韩 is written “Hang”, in Hanyu Pinyin “Han”, and Cantonese “Hon”. The negation of a relationship between a pair of distinct proper names is an important element as string distance algorithms cannot discriminate between similar names accurately (see Figure 4.10.). Therefore, other methods including the Proper Name Classifier (see above) and using secondary sources are valuable in this discrimination.

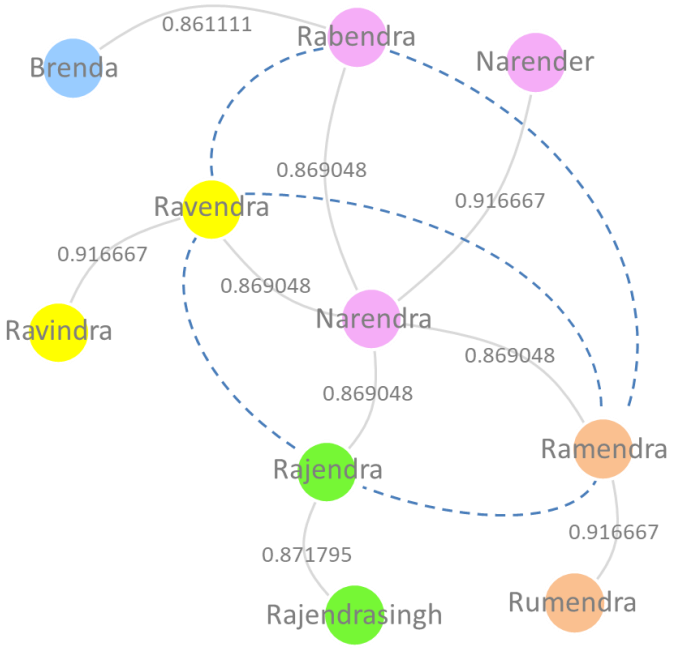


Figure 4.10. This figure illustrates a part of the Given Name Reference Graph (GNRG) with annotated blue dashed lines indicating where the secondary source data has negated a relationship between pairs of proper names.

Figure 4.10 represents how the four proper names “Rajendra”, “Ravendra”, “Rabendra”, and “Ramendra” are discriminated between using the secondary source data negating these relationships.

The next step involves using the Louvain algorithm (Blondel et al., 2008) to partition the graph of proper names into non-overlapping communities or classes. The Louvain algorithm was deployed here for the granularity of communities and the relative runtime speed. An illustration of how the community detection results in classes is based on the colour of the nodes in the GNRG in figure 4.10 and in the FNRG in figure 4.11.

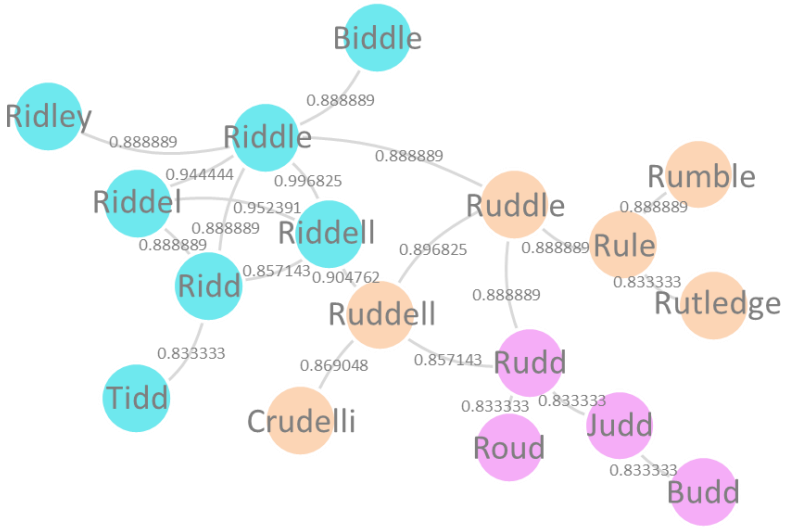


Figure 4.11. This figure depicts a portion of the Family Name Reference Graph (FNRG), with node colour representing the blocking classes derived through community detection.

An additional feature of the reference graph is that the frequency of names that each blocking class represents is retained as an attribute. This allows the block sizes to be determined very quickly enabling optimization.

The basic output, in conjunction with a reference graph, is a table of nodes with a corresponding integer or a node integer attribute. When applying this metadata for blocking purposes a number of options are possible, ranging from a simple integer scalar representation through to the more complex representation of a ragged array of integers when dealing with multiple name scenarios, whether Given name or Family name.

RGA Performance

Experiments conducted concluded that whilst many blocking algorithms were computationally efficient the accuracy was a severe limiting factor to the overall accuracy of the ER model. In response to this the RGA was developed. We will now systematically discuss the experimental results (see table 4.7) of the RGA in relation to runtime, scalability, and accuracy using the four evaluation datasets, comparing RGA against two other blocking algorithms (Metaphone3 and 1st, 6th, 9th letter (169) algorithm).

Table 4.7. Experimental Results of the Reference Graph Algorithm: Computational expense, accuracy and scalability.

Evaluation of the Reference Graph algorithm

	Sanctions *RGA | Phonetic | Trunc	**Dark Network / STR** RGA | Phonetic | Trunc	**Offshore Leaks** RGA | Phonetic | Trunc	**NZ Companies Office** RGA | Phonetic | Trunc
Data				
Vertices	~23,000	~360,000	~1.4 m	~16 m
Edges	~44,000	~900,000	~2.4m	~90 m
Persons	~14,000	~100,000	~400,000	~7 m
Organisations	~7,000	~15,000	~640,000	~4 m
Scalability				
# blocks	73 | 1,735 | 2,132	336 | 3,230 | 5,455	684 | 5,298 | 10,864	13,382 | 5,855 | 12,458
Mean block size	140 | 6 | 5	173 | 18 | 11	384 | 56 | 28	203 | 468 | 221
Max block size	3,086 | 178 | 246	1,483 | 1,415 | 1,438	2,545 | 3,383 | 2,209	14,989 | 24,876 | 20,755
# of computations	5.7m | 0.2m | 0.1m	11m | 7.5m | 5m	116m | 87m | 45m	7,758m | 6,527m | 4,468m
Runtime (seconds)				
Pre processing	18.01 | 0.03 | 0.09	25.55 | 0.17 | 0.25	89.14 | 2.44 | 2.89	107.71 | 42.64 | 26.36
Function	7.06 | 6.44 | 7.60	8.99 | 8.59 | 13.15	38.06 | 27.44 | 31.88	449.01 | 419.37 | 353.75
Accuracy				
# predictions	644 | 678 | 207	4,051 | 4,071 | 2,381	457 | 455 | 312	2.07m | 2.07m | 1.98m
Global transitivity	0.752 | 0.753 | 0.871	0.9995 | 0.9994 | 1	0.955 | 0.947 | 1	0.9999 | 0.9998 | 0.9999
False positives	0.02 | 0.04 | 0	0 | 0.01 | 0	0 | 0.08 | 0	0 | 0 | 0

* RGA = Reference Graph algorithm | Phonetic = Metaphone 3 | Trunc = 1st, 6th, 9th letter algorithm

The relative runtime of the RGA is an obvious limitation to its use, but not prohibitively so. As a percentage of overall ER model runtime the pre-processing aspect of RGA (i.e. generating the classes) remains in the 1-2% range, well in excess of the other two blocking algorithms, but still relatively inexpensive. The function runtime refers to a single execution of the Pairwise Equivalence wrapper function (see Appendix A for details). This wrapper function, dependent on parameters set and the nature of the data, will generally be executed around 30 times within the ER model. So, the distribution of the blocks and the resulting efficiency and reduced runtime of the wrapper function is an important consideration as well. We can see that the RGA is a little more expensive for each execution of the wrapper function across the four datasets.

Scalability is a related topic. The profile of blocks, or classes, generated by the blocking algorithms is an important ingredient to computational efficiency and in particular managing RAM. Blocks that are excessively large, due to the pairwise nature of the problem, will slide towards intractability consuming more RAM and runtime. Likewise a profile of too many small blocks will incur increasing latency cost. So, from an efficiency optimisation perspective the optimal block distribution would be a smallish number of similar size blocks that balance scalability and speed. However, this must be achieved whilst maximising accuracy. Every input vector of proper names is different so the generalisability of the algorithm is critical. This generalisability needs to be across different size datasets, with differing heterogeneity of proper names, and differing sources of intentional and unintentional name variation. The primary metric for scalability is the maximum block size, as this is the set of entities that pairwise comparison will be drawn from. Taking the NZCO data as an example the RGA, Metaphone3, and the 169 algorithm, with maximum block sizes of 14,989, 24,876, and 20,755 respectively, will need to generate and store data on about 112, 309, and 215 million pairs respectively. In terms of consumption of RAM the three algorithms consumed a peak of 2.7, 10, and 6.2 Gb of RAM respectively. So, from these results the RGA is by far the most scalable.

Accuracy is measured by measuring the global transitivity and the number of false positives from the set of predictions generated by the single execution of the Pairwise Equivalence wrapper function. Global transitivity will provide a very good assessment of the relative prevalence of false negatives – as missing edges from prediction components are likely false negatives. The simple goal of blocking algorithms in this context is to enable maximising the number of accurate predictions whilst minimising the number of false positives. The truncation algorithm is the most accurate with no identified false positives across the 4 datasets and extremely high global transitivity scores, however the number of predictions made by this algorithm is low in comparison. The RFA and Phonetic algorithm have similar results with the RFA slightly outperforming the Phonetic algorithm in terms of accuracy. The Phonetic algorithm identified slightly more predictions in the smaller datasets (Sanctions and Dark Network) whereas the RGA identified slightly more predictions in the two larger

datasets (Offshore Leaks and NZCO). However, the higher number of false positives from the Metaphone3 algorithm accounts for most of the higher prediction rate.

The RGA blocking algorithm is relatively expensive to run (1% of runtime) however it yields impressive accuracy and when used in combination with other orthogonal blocking approaches proves incredibly successful. Other blocking approaches used within the ER model include community detection (Louvain algorithm), and ordered unique characters algorithm (e.g. "Robinson" == "BINORS").

4.1.7.4 Collective Entity Resolution (CER)

Problem

ER is an inherent relationship problem – do *i* and *j* actually represent the same real-world entity. Logically then it follows that if *i* and *j* are the same, and *i* and *k* are the same then *j* and *k* must be the same. When a set of related predictions do not follow this logic this indicates there is the presence of false positives and/or false negatives.

Goal

Transitivity and graph theory more broadly can thus be used to improve performance in three ways. Firstly, efficiently identify these non-transitive prediction components from the Prediction Graph (a set of related predictions that contain false positives and/or false negatives). Secondly, support the contextual decision-making of how to treat the presence of false positives and/or false negatives within these non-transitive prediction components. Thirdly, generate global metrics for ER performance.

Purpose

The purpose of CER is to significantly enhance the accuracy of the ER model within an acceptable runtime and not create a prohibitive scalability problem. In terms of accuracy it is not just improving accuracy it is enabling the identification of the more complex and latent pairs of entities that cannot otherwise be identified. Focusing on this complex latent subset is a crucial element for many domains that are characterised by complexity and intentional identity obfuscation (e.g. policing and counter-terrorism).

Design

This sub-module is designed to take the pairs generated from the previous sub-modules (via pairwise approach), the Prediction Data, and treat this data representation as representing all of the entities that are potential duplicates. This philosophical position changes the focus from the continuance of attempting to find unobserved complex equivalent pairs within the entire data, or problem space, to the narrower problem space anchored by the set of predictions (both validated and invalidated) and associated metadata discovered thus far. Focussing on the set of predictions, and associated metadata, already generated creates the opportunity to represent these predictions as a graph and use contextual graph topology metrics in concert with the predictions as a basis to identify additional predictions and stringently assess those predictions that are coupled to the highest uncertainty.

It is important to note that the transitive closure and exclusivity are contextualised so better decisions are made. It is not simply a case of relying on one method (such as transitivity) to blindly make additional predictions. Transitivity is used to identify potential false negatives and the context of the pair, and surrounding predictions are used to make a decision. Likewise exclusivity is used to identify potential false positives and the context of the pair, and surrounding predictions are used to make a decision.

The sub-module is comprised of four parts:

- Specific vector-based battery of functions targeting specific scenarios,
- A transitivity algorithm designed to infer the presence and absence of edges, or in other words uses transitivity to identify potential false negatives,
- An exclusivity algorithm designed to identify contextual logic irregularities and invalidate previously predicted matches, or in other words uses transitivity to identify potential false positives,
- A machine learning algorithm (RPART or SVM) to classify matches by generating a probabilistic score [0-1],

Why is context required?

Context is fundamentally critical to making a decision as to when a pair of entities in fact both represent a single real-world entity. Four elements are particularly important, including:

- taking relational data into account,
- making entity resolution decisions in the context of associated entity resolution decisions,
- explicitly acknowledging and incorporating the context of why the entity resolution is being undertaken in the first place, and
- explicitly measuring the models performance and incorporating this into decision making.

Making a decision on whether two entities are in fact the same real-world entity is a contextual decision, and cannot be made in a vacuum that only contains node attributes. There is the relational data that gives additional information to support decision-making, such as when a pair of nodes both reside at the same address, or both are associated to one or more of the same people. Many novel algorithms have been discussed in the literature (e.g. Bhattacharya & Getoor, 2007) however the computationally inexpensive approaches of community detection to indicate whether the pair of nodes are members of the same community, and neighbourhood membership, measuring whether the pair share the same graph neighbourhood, to the order of *k*, are very effective. Their effectiveness is significantly increased within this model as the raw graph is contracted twice. Firstly, following the Deduplication sub-module, designed to only identify "certain" matches (i.e. pairs with a lot of information and exact matching), and then secondly subsequent to the Obvious Resolution sub-module, which is designed to focus on "near-certain" matches (i.e. pairs with a lot of information and very close to exact matching, or slightly less information and exact matching). As the original graphs accuracy is enhanced the accuracy of notions of graph distance can be deployed more effectively.

Contextual decision-making in entity resolution leans on two key concepts - transitivity and exclusivity. Transitivity is the notion that if *i* is equivalent to *j* and *j* is equivalent to *k* then logically *k* must be equivalent to *i* – a transitive graph structure (see Figure 4.12. for an example). Therefore, any non-transitive structural patterns generated by pairwise equivalence must either contain one or more false positives or contain one or more false negatives. Exclusivity is the notion that if *i* is equivalent to *j* and *j* is equivalent to *k* and *k* is not equivalent to *i*, then either, or both, *i*-*j* and *j*-*k* cannot be equivalent. Transitivity and exclusivity are key components of the Collective Equivalence Resolution sub-module.

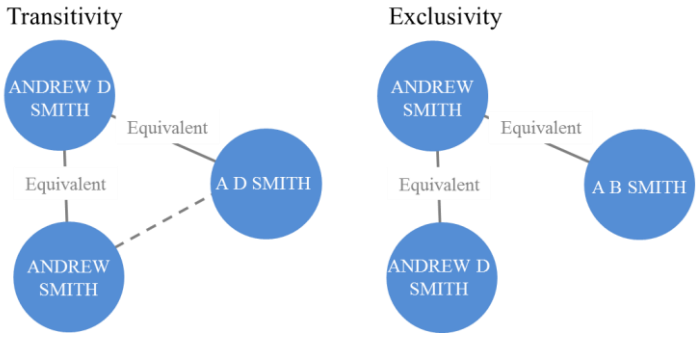


Figure 4.12. This figure gives fictitious examples of transitivity and exclusivity.

Lastly, the explicit measurement of the models performance is critical to support transparent evidence based decisions. The uncertainty derived from both data quality and model deployment is important to make visible in an accessible way so decisions can be traced to source, whether for the purposes of

testing, iterative improvement, automated analytic decision-making (e.g. to release a refund or not), or the judicial process.

Framework of how this is achieved

This sub-module uses the Prediction Data from the Obvious Resolution and Non-Obvious Resolution sub-module, which as stated previously is designed to capture close to all true positives and as a consequence many false positives. The Prediction Data is the basis from which a battery of specific vector-based functions are deployed to uncover latent knowledge, particularly where there is a lack of information available. This is possible as the vector-based approach constrains the problem space by an order of magnitude over the initial problem space defined by pairwise operations. This therefore enables the use of more computationally expensive operations. For example, in the Offshore Leaks the ~200,000 person entities identified equates to close to 21 billion operations in a pairwise problem space whereas subsequent to performing the Deduplication and Obvious Resolution sub-modules which outputs Prediction Data where the vector-based problem space is constrained to a mere 370,000 operations.

The augmented Prediction Data generated from this vector-based approach is then used to construct a graph (Prediction Graph) of ER predictions and all the non-transitive components of the graph are identified and extracted into a subgraph. This non-transitive subgraph represents the opportunities to use transitivity and exclusivity to generate new potential matches (identify and resolve false negatives) and prune existing matches represented as edges (identify and resolve false positives).

Specific vector-based battery of functions

The first facet of the CER sub-module performs a battery of vector-based operations to uncover more latent knowledge within the Prediction Data. The idea here is that there is an opportunity to focus on a specific entity resolution scenario that reflects a specific driver of duplicates, and develop a very precise approach to provide metadata to make better decisions on this specific problem subset. As the ER model matures and further ideas are developed there remain slots available to easily slot in new vectorised approaches. These could be very specific approaches to deal with ER problems that are specific to different parts of the world, or dependent on the data being resolved.

Arabic name origin. The first vector-based function takes the person entities from the Prediction Data (17,000 in the Offshore Leaks example) and targets names from Arabic name origin. Assessment is made via the following steps:

i. Identify the subset of person matches from the Prediction Data by focussing on instances where the source or target person's name is from the Arabic name origin.
ii. Conduct semantic based cleansing based on the target set. For example, names of Arabic name origin may or may not include the nasab, a patronymic or series of patronymics, indicating the entities lineage. Therefore, the harmonisation of names by ignoring the nasab (for example, "IBN", "BIN", "BINT") improves performance.
iii. Conduct ASM algorithms (Cosine, Jaro-Winkler, and Jaro-Winkler (p=0.2)) with a 'strict' threshold setting.
iv. Use the Proper Name Classifier to eliminate false positives deterministically (leveraging off the set of Arabic names within the onomastic gazetteer).
v. Utilise the ASM algorithm results to generate a rule based approach (using a conservative threshold tailored to this name subset) on which pairs to add to the Prediction Data, as additional potential matches.

Initial detector. The second vector-based function focuses on pairs from the Prediction Data which have initials. Assessment is made via the following steps:

i. Identify the subset of person matches from the Prediction Data which contain at least one given name initial.
ii. A logic vector-based approach is then used to compare the source and target name, including potential hypocorism utilising the Hypocorism graph, for logical inconsistencies.
iii. A tuple based approach to compare all name elements (i.e. each atomic name in the name of the source node to every atomic name in the name of the target node) using the Jaro-Winkler ASM algorithm.
iv. Use the Proper Name Classifier to eliminate false positives deterministically.
v. Utilise the ASM algorithm results to generate a rule based approach on which pairs to add to the Prediction Data, as additional potential matches.

Name transposition. The third vector-based function focuses on pairs from the Prediction Data which have transposition of given names. Assessment is made via the following steps:

i. Identify the subset of person matches from the Prediction Data where the source and target contain equivalent DoB.
ii. A logic vector-based approach is then used to compare the source and target given names using the Given Name Reference Graph (GNRG).
iii. The GNRG membership of each given name of the source and target is generated and the intersection is measured (e.g. (7, 149, 38) to (149, 7)) and compared to the maximal possible intersection length (e.g. 2).

iv. Those pairs which have maximal intersection of given names are then added to the Prediction Data, as additional predictions.

Contextually applied transitivity (transitive closure)

Contextually applied transitivity is based on taking each non-transitive component from the Prediction Graph – as these contain predictions that are false positives and/or false negatives – and providing specific metadata under this context about each pair in the component. Subsequent to the metadata generation a decision is made on which transitive edges to add (i.e. resolving false negatives).

Each non-transitive component within the Prediction Graph is augmented, as edge attributes, with the following metadata across multiple attribute tuples. In this sub-module name attributes are represented in their atomic state as tuples (e.g. {"Robinson", "David", Derek"}), rather than as a single string where atomic names are concatenated (e.g. "Robinson David Derek"). This tuple based approach sacrifices computational expense for increased specificity of metadata generation enabling more specific logic usage. As the problem set has been constrained significantly now focussing on very specific instances this approach is now both plausible and desirable.

- String Distance [0-1]: a numeric measure where 0 is dissimilar and 1 equivalent, utilising the ASM algorithms Jaro-Winkler, Cosine, and Longest Common Substring.
- Social Distance [0,1,2]: this is an additive binary amalgam of Neighbourhood Distance [0,1], indicating whether the source and target node are neighbours (order of *k*), and Common Community [0,1], indicating whether the source and target node are within the same community as measured following the second graph contraction on the raw data. Social Distance was constructed as a higher level concept creating the benefits of reducing the computational expense and simplifying the model. Social Distance is a robust metric indicating whether source and target node is proximate in the raw graph.
- Name Frequency [0-1]: a numeric measure where 0 is unique and 1 is the most common name. The Name Frequency Algorithm is deployed as a computationally efficient approach that accepts name words as independent (although they are clearly not) and measures how unique they are relative to all other name words. Importantly the algorithm is originally seeded by taking only original names so not to generate bias created due to duplicate names. For example, within the Offshore Leaks "GAETANNE" is noted as a name on 28 occasions, however these instances all relate to the same person and so is counted as 1.
- ER_Rule: a character element that identifies which wrapper function was used to make the prediction, including wrapper functions from the Deduplication, Obvious Resolution, and Non-Obvious Resolution sub-modules and the vector-based functions above.

- Local Transitivity [0-1]: where 0 is perfectly non-transitive and 1 perfectly transitive. The metric is based on taking the mean of the source and target nodes local transitivity scores. This metric gives insight as to how transitive the local surrounding component is.
- Information Quantity [0-1]: where 0 is a severe lack of information and 1 represents high quantity of information. This metric is constructed by measuring how many elements (e.g. Family Name, Given Names, DOB) have data available to enable assessment.
- Onomastic similarity [0,1]: Detects the presence of a hypocorism within the pair.
- Resource Allocation Index [0-1]: The Resource Allocation Index (RAI) (Zhou, Lu & Zhang, 2009) is used to determine which pairs, with no existing edge (prediction), may potentially represent the same real-world entity. The RAI works by measuring the product of the normalised degree of all nodes along every shortest path between a pair. This metric has computational limitations when implemented in a matrix based or native graph algorithm, however when implemented in a table based design we generate computational improvements in the same vicinity as we did when we rewrote the graph distance (shortest path length) algorithm (see above). Furthermore, pair generation for input into the RAI function is constrained by a neighbourhood distance with an order of k (default is 4) which provides another mechanism to ensure computational efficiency whilst retaining high performance.
- Proper Name Origin Classifier: The Proper Name Origin Classifier indicates what the origin is of the source and target nodes atomic names. Doing this can establish enhanced context. An example in which this is used is identifying anglicised atomic names, which may indicate (if the family name, DoB, and possibly one of the given names are present) whether the least transitive vertex of a non-transitive prediction component is equivalent but using an anglicised given name (see Figure 4.13.). In the example demonstrated in figure 4.13 we have a non-transitive prediction component, illustrated by the blue nodes. We can see that name similarity and a shared address has created enough context to make the decision that Wayne Wang and Wayne Chih Yu Wang are predicted to be the same real-world entity. But there is not enough information available to establish that Chih Yu Wang and Wang Chih Yu are also the same real-world entity as Wayne Wang. The result of this is a non-transitive component. Using the PNOC however we can identify that Wayne Wang is an anglicised name and thus creating additional information to make the assertion that Wayne Wang is likely an anglicised variant of Chih Yu Wang, Wang Chih Yu and Wayne Chih Yu Wang.

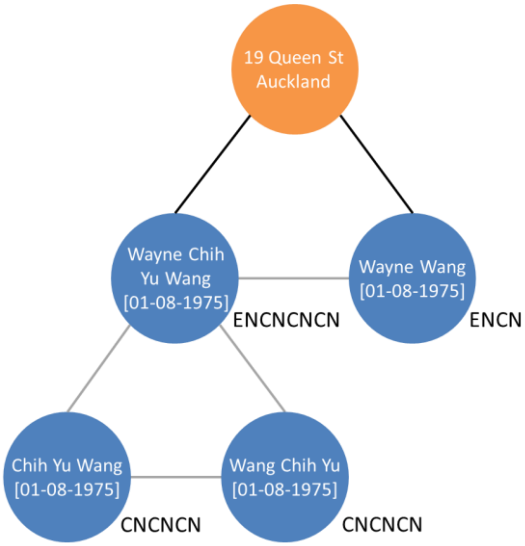


Figure 4.13. Fictitious example of how the Proper Name Origin Classifier can establish context.

- Proper Name Classifier: The Proper Name Classifier is used to deterministically prune false positives.

So, transitive closure is used to identify a set of new edges (predictions), then the Proper Name Classifier is used to prune those newly identified set of edges where the source and target node contain proper names that are different and not likely generated by error. Subsequently, a series of specific scenarios are then implemented in the form of a battery of set of rules which select which new transitive edges are added to the Prediction Graph, or in other words which transitively derived predictions are added to the Prediction Graph. Pairwise rules include:

One of the source or target nodes has a given name that is a hypocorism; AND the remainder given names are transposed; AND the DoB is not different; AND the family names are near exact.

One of the source or target nodes has a given name or given names that contain logically consistent initials; AND the remainder given names are exact; AND the DoB is not different; AND the family names are near exact.

The source and target nodes have a small graph distance; AND the given names are near exact; AND the DoB is not different; AND the family names are near exact.

One of the source or target nodes has a compound family name which one part is an exact match; AND the given names are near exact; AND the DoB is not different.

One of the source or target nodes has a anglicised given name; AND the remainder given names (if any) are near exact; AND the DoB is not different; AND the family names are near exact.

These rules select the set of new "Transitive" edges (ER predictions) are added to the non-transitive subgraph.

Contextually applied exclusivity

Contextually applied exclusivity is based on taking each non-transitive component from the Prediction Graph (including those newly identified transitive predictions) – as these contain predictions that are false positives and/or false negatives – and using the specific metadata under this context to make a decision about which non-transitive edges to assign as invalid (i.e. resolving false positives).

Exclusivity is applied by taking the non-transitive subgraph, augmented with transitivity, and accepting the following as validated matches:

Predictions derived from high certainty functions (i.e. models with high transitivity, information quantity, and a high percentage of predictions validated by other functions) or in other words high probability predictions.

Predictions within a completely transitive area of the component – i.e. where edge local transitivity equals 1.

Those matches where there both the source and target share the same graph community.

Those matches where the edge local transitivity metric > edge local transitivity transitional point (e.g. 0.8) and the Name Frequency is probabilistically uncommon based on the variance of the distribution, indicating a near completely transitive area of the component involving uncommon names.

The edge local transitivity transitional point is derived through performing a simple k-means clustering on the density of the distribution of the local edge transitivity and local edge transitivity to find the best transitional point between clusters. Applying the rule in this manner identifies instances where the failure to attain complete transitivity is likely to be the failure of the model to detect the

edge rather than a genuine example of exclusivity. Testing proves this to be the case. A fictitious example based on the Offshore Leaks demonstrating this is provided in figure 4.14. below. The upper left-hand pane displays a highly transitive component of nodes with an uncommon set of names (the grey edges indicate the missed edges). The right-hand pane displays a highly transitive component of nodes with a common set of names. The lower left-hand pane displays the frequency of local edge transitivity, with the colours depicting the two clusters, making the transition point 0.8. Further testing is required to test the wider generalisability of this method.

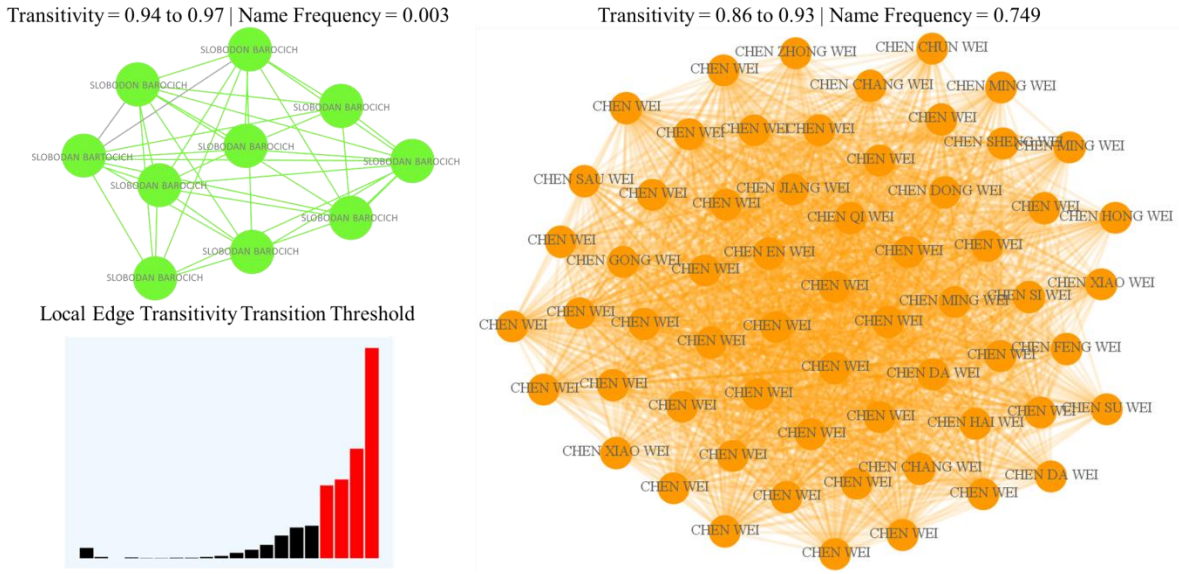


Figure 4.14. This figure gives fictitious examples based on the Offshore Leaks of how edge transitivity and name frequency are used to decide which non-transitive components are accepted as representing equivalent real-world entities. The upper left-hand pane displays a highly transitive component of nodes with an uncommon set of names (the grey edges indicate the missed edges). The right-hand pane displays a highly transitive component of nodes with a common set of names. The lower left-hand pane displays the frequency of local edge transitivity, with the colour depicting the two clusters, making the transition point 0.8.

> Those matches where there both the source and target node have at least three name words each and the edge local transitivity metric > the edge local transitivity transitional point (e.g. 0.8).

This indicates a near completely transitive area of the component involving complex names. Therefore, the failure to attain complete transitivity is likely to be the failure of the model to detect the edge rather than a genuine example of exclusivity – which is proven by testing. A fictitious example based on the Offshore Leaks is provided in figure 4.15. where predicted edges are shown in blue and the missed edges are in grey.

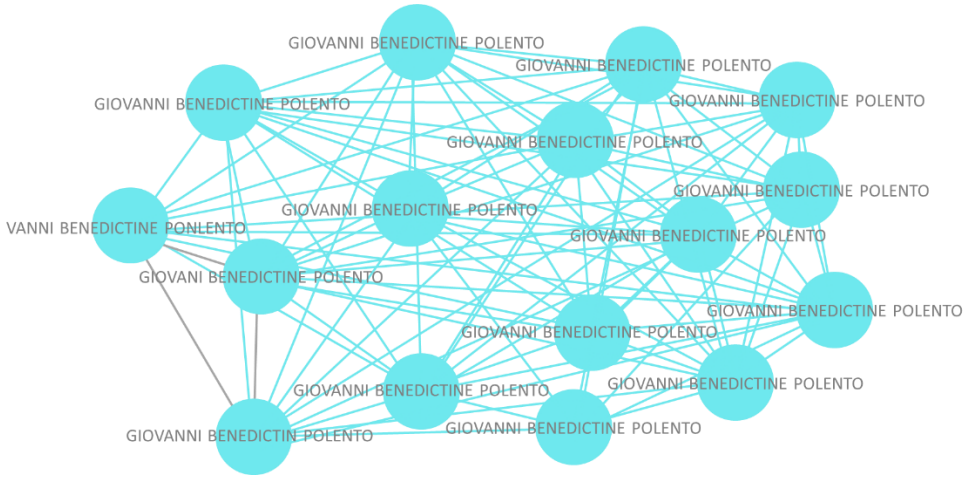


Figure 4.15. This figure gives a fictitious example based on the Offshore Leaks of how edge transitivity, name frequency and complexity of name are used to decide which non-transitive components are accepted as representing equivalent real-world entities.

Performance of the Collective Entity Resolution sub-module

The performance of the Collective Entity Resolution sub-module is illustrated in table 4.8 below which contrasts the runtime (in seconds) of the sub- module in the context of the ER models overall runtime, the global transitivity of the Prediction Graph before and after Collective ER, with the total number of predictions also listed. It was not possible to compare the results derived from the ER model using the CER sub-module with the ER model not using the CER sub-module. This is because the design premise of the overall model is to initially identify the set of predictions that indicate a possible 'match' and then use CER to generate additional metadata to support decision-making on the most uncertain and complex predictions. Simply not running the CER sub-module results in significantly poorer accuracy as expected.

Table 4.8. Illustrates the performance of the Collective ER sub-module across the four evaluation datasets.

Computational expense & contribution to accuracy of Collective ER (CER)

	Sanctions	Dark Network / STR	Offshore Leaks	NZ Companies Office
Data				
Vertices	~23,000	~360,000	~1.4 m	~16 m
Edges	~44,000	~900,000	~2.4m	~90 m
Persons	~14,000	~100,000	~400,000	~7 m
Organisations	~7,000	~15,000	~640,000	~4 m
Runtime (seconds)				
CER module	154	536	842	10,162
Overall ER model	339	1,439	3,083	48,537
Proportion	0.45	0.37	0.27	0.21
ER model performance				
Global transitivity pre CER	0.693826	0.999905	0.99921	0.999647
Global transitivity post CER	0.745456	0.999989	0.99953	0.999895
Number of predictions	97,429	646,111	1,774,502	21,081,313

As expected the global transitivity improves across all four datasets, with the data that contains more complex duplicates (Sanctions) improving the most. This is expected as Collective ER focuses on the most uncertain true/false positives and associated false negatives, identified by detecting the non-transitive components of the predictions made (i.e. the Prediction Graph). So, data that contains more uncertainty will benefit most from Collective ER and conversely data that contains less uncertainty will benefit less. So, it is clear from these results that the implemented Collective ER sub- module significantly improves performance, particularly on complex non-obvious data, and scales in a sub-linear fashion relative to the remainder of the ER model's sub- modules.

4.1.7.5 In situ ER prediction

Problem

The contextual application of ER predictions to the raw data is often ignored. ER models generally treat each entity within the input data as an isolated node which is compared to all other isolated nodes of the same type to derive some understanding of which pairs are sufficiently similar to be classified as a match. But what about the contextual change in the data when the ER predictions are applied? There is new data context, post the materialisation of ER predictions, which enables better decision-making.

Goal

The goal is simply making use of the newly created data context, with ER predictions added, to create more knowledge. This goal has been realised through the creation of an algorithm that assesses the contextual relevance of the ER predictions.

Purpose

The purpose is to improve certainty in relation to data quality, which has a significant flow through impact onto downstream decision-making. For example, when attempting to identify criminal subgraphs the quality of the ER predictions is critical to whether the extracted subgraph is relevant or not.

Design

There is no literature found that attempts to use the contextual data, with ER predictions applied, to generate more knowledge on ER performance and the subsequent impact on making use of the newly fused data.

The CER sub- module takes the context of the graph, local clusters and nodes when making supporting decisions on what predictions are false positives or true positives. Machine learning takes the broader context of the entire metadata set including the metadata of function parameters and performance when making a classification prediction. The opportunity is then presented to create more knowledge about the ER applied performance when applying the ER predictions to the data. The design of this feedback mechanism is an algorithm that uses graph metrics and contextual (edge and vertex attribute) metadata to mimic human decision-making.

The in situ algorithm uses the context of when these predictions have been applied to the original data to generate contextual understanding of prediction likelihood. Figure 4.16 represents an example. The pane on the left represents an instance of where a number of datasets have been entity resolved – the entity resolution predictions are represented as red links. So, you can see a grey, pink and cyan node predicted to actually represent one real-world entity, connected by the red links. There are no other entity resolution predictions. The pane on the right however has three distinct pairs of entities (pink and grey nodes connected by a red line) that are entity resolution predictions. The context of this cluster of predictions makes it more likely they are all correct compared to the example in the left pane which has an absence of any other predictions. This is based on the left pane example involving only one real-world entity (represented as three separate entities in the data) whereas the right pane example includes ER predictions for three separate real-world entities - with the set of entities represented in each dataset (grey and pink) being proximal to one another. This proximal nature of sets of predictions is strong inferred evidence of ER accuracy.

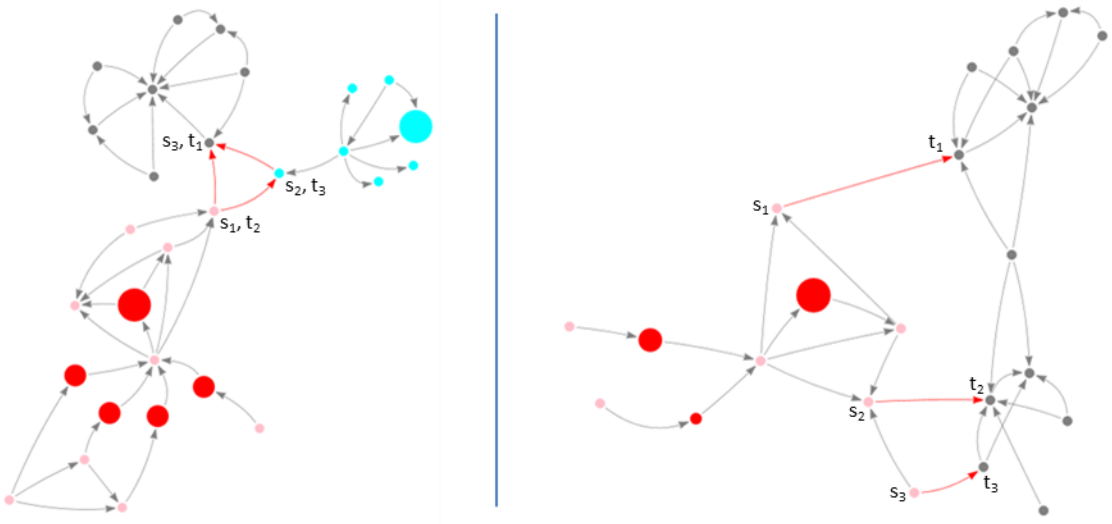


Figure 4.16. An illustration of two subgraphs with the left pane an example of three ER predictions focusing on one real-world entity and the right pane an example of three distinct ER predictions focusing on the resolution of three real-world entities.

The ‘ER In Situ Cluster Rank’ algorithm is used to assess in situ likelihood. The algorithm is based on the sum of the inverse of geodesic distance (i.e. how many hops) between all pairs derived from the set of ER prediction source nodes (e.g. {d(S_1, S_2,), d(S_1, S_3,), d(S_2, S_3,)}) and all pairs derived from the set of ER prediction target nodes (e.g. {d(T_1, T_2,), d(T_1, T_3,), d(T_2, T_3,)}), on the non-weighted graph. The set of ER prediction source nodes is represented as *S* and the set of ER prediction target nodes is represented as *T*.

$$ER\ In\ Situ\ Cluster\ Rank\ score(x) = \sum \frac{1}{d(Si, Sj)} + \frac{1}{d(Ti, Tj)} \qquad \text{Eq. (8)}$$

So, the subgraph in the left pane would score zero as the ER prediction source nodes are not connected and the ER prediction target nodes are not connected ((1/0 + 1/0 + 1/0) + (1/0 + 1/0 + 1/0) = 0). The right pane would score 3.66 as ((½ + ⅓ + 1/1) + (½ + ⅓ + 1/1) = 3.66). Effectively the more ER predictions in local areas of the graph there are the higher the score. Low scores do not infer low likelihood, however high score does infer high likelihood, therefore the algorithm is rank-based.

Performance

The ‘ER In Situ Cluster Rank’ algorithm has proven utility in ranking subgraphs with high certainty ER predictions, according to users. These subgraphs can thus be prioritised as less resource is required to manually assess the ER predictions in the context of the subgraph.

Further testing is required within a robust framework to evolve this algorithm to understand how useful it is and how best to improve, or design new methods, to optimise value. See ‘Chapter 7. Potential Extensions’.

4.1.8 ER model performance

Performance measurement in the field of entity resolution is a notorious challenge as each model is constructed in a way that requires user interaction, such as tuning, setting arbitrary thresholds, or some level of experimentation to reach near optimal levels. Therefore, it is a subjective exercise to ensure model deployments are equivalent. This has driven a conservative approach in the measurement of the ER model’s performance to eliminate any criticism levelled for failing to adopt a like for like comparison.

Furthermore, the type of output produced (e.g. graph or table) to test the quality of performance is dependent on the ER technology assessed. Most ER technology will not provide a probability on each prediction and will not present pairwise predictions where the pair is predicted to not represent the

same real-world entity (i.e. true negative, false negative). This is the second element which makes direct comparison difficult.

Taking these aspects into consideration, the testing of results was done on a blind basis so testers were masked from which results came from which model. Table 4.9 outlines the results of the performance on the four evaluation datasets comparing the use of IBM® InfoSphere Identity Insight (ranked in the top solutions for entity resolution by Gartner – October 2017), and the EntityResolution package (the R based implementation of the Entity Resolution module).

Table 4.9. This table outlines the results of the performance on the four evaluation datasets comparing the use of IBM® InfoSphere Identity Insight (left) and the Entity Resolution module (right) on ER.

Performance of the ER model

	Sanctions	Dark Network / STR	Offshore Leaks	NZ Companies Office
Data				
Vertices	~23,000	~360,000	~1.4 m	~16 m
Edges	~44,000	~900,000	~2.4m	~90 m
Persons	~14,000	~100,000	~400,000	~7 m
Organisations	~7,000	~15,000	~640,000	~4 m
Runtime (seconds)	457	1,500	**2,460** | 3,083	57,620
RAM consumed (Gb)	0.25	1.8	3.5	22
ER model performance				
Global transitivity	- | 0.7487	0.9611 | **0.9999**	0.9153 | **0.9999**	- | 0.9999
Diameter metric	- | 0.9054	- | 0.9990	- | 0.9950	- | 0.9752
Precision*	- | 0.9804	0.9023 | **0.9804**	0.8853 | **0.9792**	0.9901 | **0.9984**
Recall*	- | 0.9804	0.9564 | **0.9901**	**0.9923** | 0.9792	0.6072 | **0.9907**
F measure*	- | 0.9804	0.9286 | **0.9852**	0.9357 | **0.9792**	0.7527 | **0.9945**
Entities contracted	- | 8,321	**92,990** | 88,831	**81,882** | 70,063	- | 7,366,952

* Sampling used to estimate accuracy

Sanctions is a small highly detailed data set focusing on global organised criminal and terrorist entities. As such there are many instances of using fraudulent identities and intentional obfuscation of identity, and any associated entities (address, phone, email, organisations). Nicknames and aliases are highly prevalent. Generally the data available is rich and detailed (e.g. most persons have DoB).

Dark Network/STR is a mid-size transactional dataset focussing on suspicious transactions and criminal entities. The transnational focus of this dataset means there is a more global focus and a greater focus on bank accounts, organisations, and addresses.

Offshore Leaks is a mid-size dataset that contains a global perspective with a heavy focus on organisations, and the presence of supernodes – a few nodes with many connections. The data completeness is low (e.g. persons do not have DoB) and error is high due to the nature of the data and the manual curation methods used to extract the data (e.g. persons names are presented as a string rather than having family and given names explicitly identified).

NZ Companies Office is a large dataset that contains a heavy focus on organisations, and the presence of supernodes. The data completeness is low (e.g. persons do not generally have DoB) partly due to the lack of data validation when companies are created.

Each dataset places a different focus on performance, testing the ER model in a robust way from a range of angles.

Each set of manual validation results producing the Precision, Recall, and F-measure, were generated using a sampling process as outlined above in the Output, Measurement, Diagnostics and Visibility section. This sampling process generated a set of 562 pairs for the Offshore Leaks data, 232 pairs for the Sanctions data, 1,615 pairs for the Dark Network data, and 1,414 pairs for the NZCO data. These pairs were presented in a contextual visualisation (see figure 4.3 for an example) for manual assessment from analysts. As there is no 'ground truth' there is a reliance on the judgement of the analyst.

The EntityResolution package has been designed to identify a random sample of positive and negative predictions. This means that half of the sample contains the most uncertain predictions which gives an enhanced understanding of performance to any user but also creates a biased sample to generate the approximate F-measure from. This is a design feature, however the two comparison ER solutions did not have this feature, meaning that the F-measure samples for the EntityResolution package are very conservative, whereas the comparative ER solutions are not. Even ignoring this disparity, the EntityResolution package significantly outperforms the competitors consistently. The easiest way to understand the accuracy differences is that, for example, when resolving the Offshore Leaks the EntityResolution package will make an estimated 208 errors for every 10,000 predictions versus an estimated 643 errors for every 10,000 predictions for the commercial software.

The results clearly indicate the Entity Resolution module is significantly more performant from an accuracy perspective, most notably in terms of the absence of any real cost trade-off between Precision and Recall. This reflects the non-linear relationship between Precision and Recall when using the Entity Resolution module due to the Collective Equivalence Resolution sub-module. The cost for this improved performance is runtime, however in many applied domains this runtime difference will not outweigh the accuracy benefits. Runtime is one aspect to the ability of the user to deploy the ER model in a suitable timeframe. The other is the amount of experimentation required to optimise the model for a specific dataset. As the Entity Resolution module provides automated performance metrics and a sampled prediction output, the assessment of the model's performance can be done quickly and therefore the model can be deployed in a faster turnaround.

Of note, is the poor performance of the commercial software in resolving the Companies Office graph. This was largely due to the inability of the commercial software to deal with detecting duplicates when there is very little node attribute data to generate a key with – often only given names and family name, and a non-standardised address. This highlights the value, particularly in 'low signal' applications, of using the contextual data provided by a graph to support decision making.

The EntityResolution package is significantly slower in terms of runtime, which is expected as it is developed in the R language, however the computational specs have a significant influence. For example, running the EntityResolution package on an AWS platform with 16 cores rather than 8 cores and running a clock speed of 2.7GHz rather than 2.2 GHz results in a runtime of ~12 hours (versus ~16 hours) when applied to the fused graph.

From these applied results the EntityResolution package is a widely applicable ER model that is significantly more accurate than competitors, however this accuracy improvement comes at the cost of runtime, and potentially scalability. However, it must be kept in mind that the EntityResolution package is coded in R, and significant performance improvements would be expected if engineered into a production ready state.

Robustness

The Entity Resolution module is structurally deterministic in terms of how it generates the features for the input to the SVM model. However, a key non-deterministic element is the RGA blocking. When deployed on big data the RGA uses a sampling technique to generate estimated optimal partitions, and therefore small differences in the output from the RGA are produced. This repeatability variance translates into small but significant differences when measuring ER performance. For example, differences for ER on the NZ Companies Office data would consistently range between 200 and 500 vertex contractions in the context of a total 7.4 million vertex contractions. This level of difference can translate into a very small difference in F-measure and a difference in global transitivity of around 0.00002.

Generalizability

With a significant amount of effort ER models can be tuned to produce very good results for a specific dataset or problem, however often these are by definition over-fit to the specific instance and therefore not generalizable. This model has only been tested on four datasets and performance needs to be tested on further datasets with a more global flavour to enable a more rounded understanding of generalised performance. Specifically, address and person names can be specific dependent to the global region the data represents.

However, the goal of this model was to construct an extensible generic framework that enables iterative development of generalizability, as we come across specific sub-problems in relation to entity resolution. Importantly, we treat entity resolution as an umbrella term, that merely represents a collection of specific causal mechanisms that are reflected in varying degrees across differing datasets. At this point it is useful to remind the reader that the broad goal was to provide a model that generated more accurate results than commercial products, acknowledging that there was likely to be a trade-off with runtime.

So, the founding concepts are generic, the code is written to allow for significant variance in data, the functions hierarchically constructed into wrapper functions so all that is required is ETL, an understanding of business requirements so the appropriate Tolerance level can be set, and being able to validate performance appropriately.

4.1.9 Conclusion

The problem of having real-world entities represented multiple times in the data representation is common. Within the criminal domain the complexity of this problem increases due to the intentional misinformation generated by criminal entities obfuscation. The fusion of multiple datasets only increases the number of duplicate entities (e.g. the intersection between set A and set B) and complexity of solving this problem. Entity resolution has been developed to solve this problem, however the deployment of traditional entity resolution technology to real-world criminal applications has limited utility. This limited utility is based on a lack of accuracy. The ER model developed here is designed to significantly improve generalisable accuracy and remain scalable and fast enough to retain utility in real-world criminal applications.

The evaluation of the developed ER model thus far indicates that it outperforms the commercial software competitors on accuracy with an average Precision, Recall and F-measure of 0.9860, 0.9867, 0.9863 versus 0.9259, 0.8520, and 0.8723 respectively. The accuracy advancement is especially significant in ‘low signal’ duplicate detection - detecting duplicates when there is very little node attribute data. The ability to detect these ‘low signal’ duplicates is due to the iterative contextual graph-based approach.

The computational efficiency and scalability of the model indicates utility in graphs up to a size of ~18 million nodes and ~93 million edges with a runtime of around 12-16 hours on a low spec non-distributed architecture on the fused graph. This performance is far from prohibitive for many criminal domain applications. It is however important to couch these performance metrics in the context that the code is not engineered for production performance, nor deployment.

In terms of robustness the ER model is fundamentally deterministic with some very small variation over repeat use generated through sampling routines on large data inputs. The generalisability stands up well when compared to the performance of the competition when compared across the four evaluation datasets. Generalisability is very important when ER is deployed for fusing multiple heterogeneous datasets, as the deployment of poorly generalised ER technology for data fusion purposes will unavoidably generate pockets of poor performance creating significant error and bias.

The ER model has been developed into a wrapper function written in R and contained within the closed source R package EntityResolution.

4.2 Link Prediction model

Subsequent to the fusion of data via entity resolution we can now focus on missing links, a subset of data incompleteness. Links are relationships (or edges) explicit in the data that for example could be shareholder relationships, linking people to companies, transactional relationships, linking bank accounts, or more generic relationships like "associated to", linking people to people. Missing links are those relationships between people that exist in the real-world but are not represented in the data. Data in the criminal domain, as discussed earlier, is a partial data representation of the real-world, and as such there is an opportunity to use the resolved data and link prediction technology to probabilistically identify missing links, improving the quality of the data.

Not all edges are equal. The value of identifying strong ties, edges that exist in dense parts of the graph between pairs of entities that have a graph distance of two, is not nearly as useful as identifying weak ties, being those edges that connect sparse areas of the graph where the pairwise graph distance is greater than two or where the pair of nodes exist in different components. However, weak ties are much more difficult to predict due to the reduction of available information and the intractability of comparing all pairs in a graph.

So, the challenge is using inference and prediction to identify the set of edges that are of most value and accurate, at speed, to improve the quality of the data and enhance downstream knowledge discovery.

Synopsis

The Link Prediction (LP) model developed here focuses on developing a series of quality features from which a Support Vector Machine (SVM) is used to predict both "strong ties" and "weak ties" (Granovetter, 1973). Strong ties are those relationships between people that are highly clustered and derive largely from cyclic closure. Types of relationships that fall under strong ties are familial and

close friends, where high trust exists and the relationship is maintained in an ongoing way. Weak ties are those relationships that are not highly clustered and do not form through cyclic closure. Weak ties are formed through focal closure, where relationships are not developed through direct associates but from a mutual experience (e.g. attending a conference). Weak ties are of much interest for a number of reasons including, they generate a "small-world" that connects communities or clusters of strongly connected groups of people (Newman, 2001b), they tend to create access to novel resources (e.g. knowledge, material), and those individuals that maintain multiple weak ties tend to hold strong positions in the network to control the flow of knowledge. Often LP models will discover edges (strong ties) in clustered parts of the graph that are in a relative state of better completeness. The edges predicted in this situation are less likely to enhance the more incomplete regions of the graph. This is why the detection of weak ties is so important. Having said all of this it is important to acknowledge that because criminal domain problems are always limited by partial data an open world lens needs to be applied. The implications of this is that predicted edges that display weak tie characteristics may in fact be a strong tie within a cluster of unobserved/unobservable strong ties.

The LP model developed here utilises a combination of features that have been devised to measure both focal and cyclic closure. This was achieved by selecting a set of features that are homophily and graph-based. The ability of the model to predict weak ties exists in two ways. Firstly, the model uses inferred links and then builds on top of this logic-based approach by materialising these inferences and then predicting additional links utilising this new knowledge. Secondly, the model can be deployed in an iterative way to persist the high certainty predictions made in the prior iteration, making new predictions using the previously persisted edges. In this way we can utilise the metadata generated in an iterative way and predict the existence of weak ties – defined here as those edges that connect a pair of nodes beyond a path length of two.

Figure 4.17 illustrates by example how link inference (LI) and LP can be used in tandem to detect unobserved edges. Pane 1 of figure 4.17 shows the raw entity resolved graph containing companies (C), persons (P), and addresses (A), with the pair of nodes being targeted for link prediction highlighted in green with a graph distance of three. Pane 2 of figure 4.17 shows the raw entity resolved graph with two inferred links (LI) and one predicted link (LP) added. In this example the link prediction has detected the usage of an alias. From this example the incompleteness of data within the criminal domain is made clear, as is the utility of LI and LP to generate latent knowledge from which to build higher performing downstream functionality.

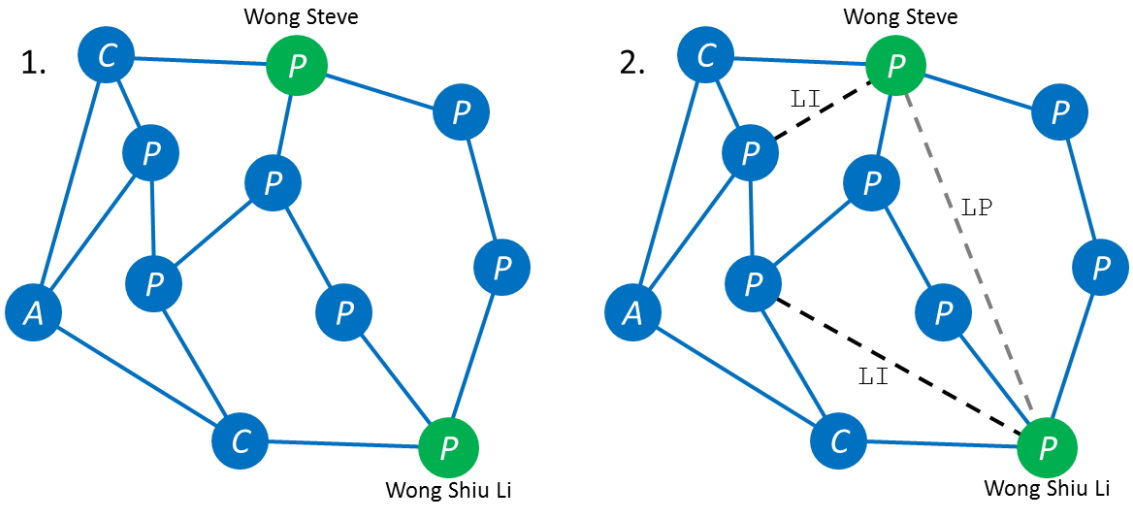


Figure 4.17. An illustration of two subgraphs with the left pane an example of a raw entity resolved subgraph and the right pane an example of that same subgraph with link inference and link prediction applied.

SVM was chosen in this instance due to the technologies impressive performance in the literature on LP (Hasan, Chaoji, Salem & Zaki, 2006), and the relative poor performance of a raft of approaches that simply will not scale (Kim & Leskovec, 2011) and / or utilise a set of assumptions, such as there are equal numbers of spurious and non-observed links, that do not apply in this domain (Guimera & Sales-Pardo, 2009). The advantages include that SVM can tackle binary classification and numeric prediction problems, and generate probability on binary classification, the compact nature of the model, and generalisability (not under or over fitting). SVM can be criticised for being a "black box" and comparatively slow however the lack of transparency is mitigated significantly by using a set of features that are underpinned by theory and the speed has not be a limiting element to date.

How was the model developed, evaluated and deployed?

The framework of how the model was developed, evaluated and deployed will now be discussed.

The development framework focused on how to understand the problem, construct the target data, engineering the optimal set of features, and tune the parameters that optimally fit the design requirements of the LP problem specific to the criminal domain.

The evaluation framework focuses on understanding the real-world performance of the LP model, through execution across the Sanctions, Dark Network, Offshore Leaks, NZCO, and fused datasets.

Deployment covers elements of how the model is deployed in the real-world, and findings in relation to its deployment.

4.2.1 Development Framework

The development framework is designed to provide a structured approach to the development of the model ensuring that the model derived is fit for purpose and explicitly tied into the concepts that underpin the problem.

Firstly, it is important to outline exactly what the LP model is not. This will serve to give insight into why modelling decisions were made. This LP problem does not assume that all nodes and edges are present, nor does it assume that the quality of the data is perfect. We are not aiming to predict future relationships, as in classic LP. The goal is to take an input graph that is a snapshot in time and predict what material relationships exist in the real-world but are not represented in the data. The definition of material relationship is important, as a relationship is an amorphous concept. Here we are defining material relationship as a relationship that confers some level of enduring trust, whether that trust is derived through mediating neighbours or directly.

I will now cover how the data was organised, how the example or target set was generated, how the pairs were selected from which to generate the features, and development model performance evaluation mimicking the reality of deployment conditions.

How was the data organised?

The model takes the entity resolved (contracted version) data as an input and goes through various critical steps to prepare the data for link prediction. The rationale for using the entity resolved graph is to build on top of the best quality version of the data, however there may well be applicability in using LP as an adjunct to ER – however this remains largely untested. Pane 1 of figure 4.18 illustrates the input data with person vertices in blue and other vertex types in black.

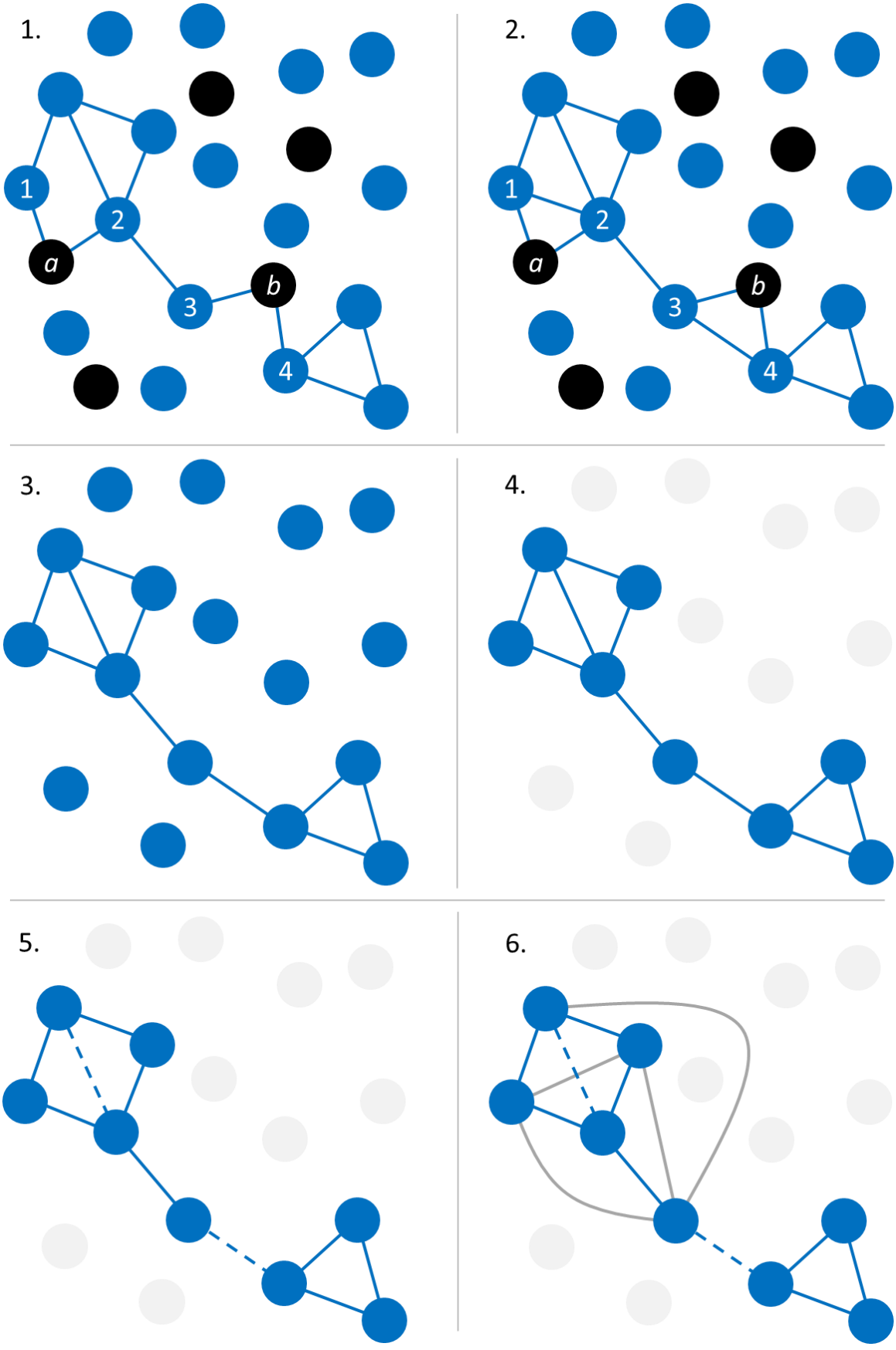


Figure 4.18. An illustration of how the LP model infers links (2), creates a person only graph (3), removes isolated nodes (4), masks a proportion of observed edges (5), and selects the example set pairs (6) ready for feature engineering.

Link inference identifies inferred missing links based on relationships to corporate entities (edges are generated between any non-supernode that maintains shareholding or directorship to a closely held company – a company with 3 or less office holders - at the same time) and co-residents (edges are generated between any non-supernode that resides in a residential property with 4 or less occupants at the same time) (see Figure 4.18. pane 2.). The notion of supernodes (defined here as any entity that has a degree > 50) is important here as many social graphs will include a small number of supernodes – those nodes that have a very high relative degree. These supernodes often represent nodes that perform intermediary roles and are characterised by a large number of non-persisting and/or immaterial relationships. For example, a casino has many transactional relationships however the vast majority of those do not infer an enduring material relationship.

Link inference is somewhat dependent on the data model, and specifically the existence of edge attributes such as edge type and edge dates to ensure the logic is rigorous. The drawback to this dependence is the increased complexity of retaining the generalisability of the LP model. Link inference nonetheless remains an important first step to improve the quality of the graph and generate more metadata for the link prediction model.

It is clear that the transitivity of the graph is related to the success of the link prediction (Murata & Moriyasu, 2007), which was also found in the testing done here. In some cases, such as bipartite graphs (e.g. Companies Office), link inference is critical in generating edges between nodes of the same class (e.g. Persons), and without link inference a traditional link prediction model would simply not be possible as there are no person to person edges available to generate a model.

Subsequent to link inference person entities were then geographically tagged, at the country level, using any associated addresses. This enables geographic attributes to be used to generate features.

The graph was then transformed from a graph with multiple vertex types (e.g. person, address, company, etc.) into a graph with a single vertex type – persons (see Figure 4.18. pane 3.). Additionally, isolated nodes, loops and multiple edges were removed (see Figure 4.18. pane 4.). This ensures that a consistent and generalizable data model is formed as a basis to approach the problem, reducing the dependence on domain-based data, and importantly the data was reduced as much as possible enabling computational efficiency.

How was the example (or target) set constructed?

The problem is fundamentally pairwise, as we are looking to predict those pairs that have a real-world relationship that are just not reflected in the data. We took all of the pairs that have edges as the

training set and then deleted a proportion of the edges (referred to as masked edges) before any feature generation. The masked edges are represented in figure 4.18 pane 5 as the dashed lines.

This is an important element of the model as the purpose of the model is to predict edges that exist in the real-world but are not included in the data. So, mimicking the missing edges by generating a set of masked edges is critical. However, every edge that we mask decreases the amount of data that we have to support the model. If we have too few masked edges then the model will be trained to find existing edges, which is not particularly helpful, and if we have too many masked edges then the accuracy of the model will erode. This problem is also linked to the class imbalance issue, generating less than ideal results. So, the hybrid example set included edges and masked edges.

On what basis were the pairs selected from which to generate the features?

The intractability of pairwise approaches is well known. So the challenge is to create a method to generate a subset of pairs from the total set of possible pairs – restricting the problem space. This was achieved by using the RAI to detect any pairs based on a shortest path length (graph distance) of 3 or under (see Figure 4.18. pane 6.). This restriction of the problem space enables other metrics that are not dependent on the graph distance between the pair – such as preferential attachment, name origin assortativity, age assortativity, and family name edit distance – to be applied in this reduced contextual problem space rather than across the whole graph. In other words predicting whether two people know each other just from age, name origin and similarity, and degree – without path or neighbour edge context – is insufficient.

Experimentation with a path length of 4 generated a significant increase in computational expense but yielded no more value. Specifically, the hop-4 model failed to identify any additional weak ties.

Subsequent to generating the list of pairs, features were extracted for each pair.

What engineered features were included for analysis?

There are two sets of features outlined below. The first set is those used within the model and the second set being those that were assessed but generated no more real value on the datasets we were deploying on. They are included because the model performance is very much dependent on the quantity and quality of the data.

Features used in the model (in order of most significant contribution).

Resource Allocation Index (RAI) [0-1]

The RAI is deployed by measuring the product of the inverse normalised degree of all nodes along every shortest path between a pair. Pair generation for input into the RAI function is constrained by a distance with an order of *k* (currently set at 3) which provides another mechanism to ensure computational efficiency whilst retaining performance. Defined as:

$$s_{xy} = \sum_{z \in N(x) \cap N(y)} \frac{1}{k(z)}. \qquad \text{Eq. (9)}$$

, where *N(x)* denotes the neighbours of *x*. This metric was used as a number of studies have found it predictive (Zhou et al., 2009; Lu & Zhou, 2009; Martinez, Berzal, & Cubero, 2017), outperforming a raft of similarity metrics including common neighbour and the Adamic-Adar index.

This metric has computational limitations when implemented in a matrix based or native graph algorithm, however when implemented in a table based design we generate computational improvements in the same vicinity as we did when we rewrote the graph distance (shortest path length) algorithm (see above). The details of the graph distance algorithm performance, recoded for an Apache Spark distributed computing context, is covered in table 4.2.

Assortativity (proper name origin) [-1:1]

The Proper Name Origin Classifier (PNOC) was utilised as the basis to generate classes of persons to measure inter and intra class assortativity (Newman, 2003a). Name origin was used as a proxy for ethnicity, as ethnic assortativity has been identified in studies (see Schelling, 1969).

Preferential Attachment

The degree of each person was calculated to establish similarity of the pairs degree, hence preferential attachment. Degree is a graph-based homophily measurement.

Age Difference

The age of the person, computed as the difference in age when viewed as a pair. Age is a node attribute based homophily measurement.

Jaro Winkler ASM of Family Name

The Jaro Winkler ASM of the pair's family name is calculated. This metric gives an indication of potential family membership, and is therefore a node attribute based homophily measurement.

Local Transitivity

The local transitivity of a node is measured by calculating the probability that a node's adjacent nodes are themselves linked. Transitivity is a graph-based homophily measurement.

Community Detection

The Louvain community detection approach was used (on the Person only graph) to generate a community partition for every person node. Community detection is a metric that is considered to generate a variation of the concept social distance, which is also measured via other metrics such as graph distance (shortest path length).

Features considered but not used in the model

Jaccard Coefficient

The Jaccard coefficient metric is a commonly used pairwise feature in link prediction due to its ease of computation and utility in predicting links (see page 23 for details).

Assortative Community Detection (ACD)

The community memberships were also measured from an assortative perspective, extending the notion of social distance to include the meta-relationship between communities of nodes.

Bridging Index

The Bridging Index refers to whether a person holds a link that bridges another community or not, based on Gould and Fernandez's (1989) approach to brokerage and Merton's (1968) notion of local and cosmopolitan influencers. It is theorised that due to homophilic reasons entities are more likely to know each other if they both hold relationships outside of their community, and are less likely if they both are limited to intra-community relationships. Bridging Index is implemented by categorising pairs as containing either two locals (0), a local and a non-local (1), or two non-locals (2). Locals are defined by nodes that only contain intra community edges and non-locals are defined by nodes that contain at least one inter-community edge.

Mutual information of topological metrics

The mutual information of topological metrics of each person pair (based only on person to person relationships graph) was generated. It is hypothesised that graph-based homophily measures will contribute to predicting preferential attachment mechanisms of relationship formation. Mutual information was the approach used to identify similarity across a range of metrics for the pair.

Betweenness

Betweenness, as described above, gives a quantified measurement of how well a node is positioned in the graph to control flow of resource. Betweenness is a graph-based homophily measurement.

Geographical proximity [0-1]

A country-based approach was taken. Each person entity was tagged with the relevant country of neighbouring address vertices. This input was taken and transformed into a pairwise approach so that a country intersection scores 1, a lack of data (i.e. NA) in either source or target vertex returns a 0.5 score, and instances where the pair of vertices has different country tags (i.e. no intersection of country tags) results in a score of 0. Geographical assortativity logically should be predictive, understanding the paucity of data available, as it stands to reason a pair of entities living in the same country are more likely to know each other than a pair of entities residing in different countries.

Feature selection

Redundancy was assessed via correlation, and importance via a determination of model performance by constructing a function that conglomerates and then prunes features. Manual assessment was then conducted to ensure the features were appropriately selected for an optimal transparent "as simple as possible" model.

How was model performance measured in development?

Applying link prediction in the context of the criminal domain places the primary emphasis on using the technology to predict real-world relationships between people that do not exist in the data. Secondarily, we can use link prediction to also predict instances where an edge exists in the data but does not exist in the real-world. We have focussed solely on the former.

So, we are looking for a high F-measure but specifically a high Precision that predicts the presence of masked edges (not all edges) well - true positives (TP) - in combination with not predicting the presence of no-edge pairs false positives (FP). However, in this context it is complicated because it is likely that some number of these FP's (and even some TN's) are in fact the very edges we are looking for – edges that do not exist in the data but do exist in the real-world. So, a human validated set of potential edges (edges that do not exist in the data) was scored by an expert for the likelihood of their existence in the real-world. This creates a second test set of 250 pairs from which to measure the model's success on an automated basis. Of course, there are difficulties with this subjective approach, as we rely on the experience of one expert, however it gives us a second dimension that reflects how the models output would be used in practice. Also, it has to be explicitly highlighted that the human

validated test set is very small (250 pairs) and that the definition of the probability of whether 'material' edges exists means different things to different people. The expert's knowledge of whether a real-world relationship exists between a pair is limited and so even with adopting an expert there will be instances where unobserved real-world relationships are predicted by the LP model but are recorded as a FP.

Utilising these two metrics (the training and test Precision and the human validation set) gives us a more balanced methodology to robustly assess the link prediction model in a real-world application setting.

K-Cross fold validation was utilised (k=10) to enable a robust cross validation error based on validation sets. This was contrasted with the training error generated and basically the optimal combination of cross validation and training error was where both these metrics were lowest, in comparison to other evaluated models. The non-optimised models built within the development framework had an average cross validation error of 0.0126 and an average training error of 0.0098 interpreted as a well fitted model.

This was sense checked in an assessment of the Precision scores for both the masked and the human validated set of pairs, with the optimal model selected for a balance of repeatability (i.e. a low performance measurement variance in repeated executions), speed and Precision of the masked and human validated set.

Repeatability is important so the SVM is deployed as an ensemble using bagging where there is a minimum of 4 iterations and 5% of edge masking, with speed considerations meaning that excessive iterations is not desirable. After experimentation a Precision score of ~ 0.998 for the masked edges and non-masked edges in the test set and ~ 0.996 for the human validated set was considered excellent given the accompanying error generated through data, modelling and human validation. So, the default model consisted of a 5% masking percentage and 4 iterations, with a cost parameter of 5. Initial performance testing indicated a statistically insignificant difference across a range of cost parameters experimented with across a variety of models. Therefore, the midpoint of the experimental cost parameters (c = 5) was set as the default for simplicity and computational performance reasons. These parameters are not hard coded but are able to be set in the wrapper function created, based on the domain context.

The difficult nature of the problem is explicitly understood and of course the domain where the model is deployed will have a range of situations that have varying requirements. In some instances the model may need to be used aggressively to detect potential leads in a case, or if the model is used to support risk model deployment a conservative approach may be preferred.

How are the FP's partitioned into true FP's and FP's that represent real missing edges?

As we are actually interested in the FP's, and perhaps even some of the TN's, subsequent to the completion of the SVM bagging approach how do we quantify which of these pairs to retain as predicted missing edges? Three options are presented to the user in terms of how to decide the cut-off for partitioning the FP scores into two classes. The default method is a percentile-based approach where the user can enter the desired percentile (default 0.995). Using a percentile-based approach makes use of the knowledge derived from the model, gives the user the ability to experiment with differing thresholds, and is quick. The second option is a mixture model with a fitted beta distribution which is used to partition the FP scores into two distributions, using a sample of normalised sum probabilities across the set of iterations. This is done in a performant way by limiting the vector of probabilities to 50,000 elements and using case weights. The drawbacks include variability of results across different datasets (in part due to sampling an imbalanced problem), a comparatively slower runtime (some minutes), and lack of robustness in terms of failing to converge. The third option is using the 0.5 probability cut-off – or in other words using the model as a classifier relying on the model to partition into TN and FP's – and utilising the set of FP's in their entirety. The metadata and probabilities are presented so the user can generate their own mechanism of utilising the predictions.

The classes generated from the above approaches are used to define between the true FP's and the set of real-world edges missing in the data.

4.2.2 LP model performance

Evaluation was focused on measuring key elements of the LP model, and to generate visibility over generalizable performance from a real-world perspective.

Robustness

The nature of the construction of the example set results in a lack of robustness (variance in repeated execution output). The lack of robustness is largely derived from the edge masking process, which makes sense as only a small fraction of edges are masked (along with the residual edges) and so each subsequent model will likely include a significantly different masked set to train the model on. Figure 4.19 displays the variance of the Precision ratio for the classification of the masked edges (blue) and a human validated set of edges (red) that were missing in the data (with some existing in the real-world and some not) across a range of differing proportion of masked edges (0.01 to 0.1) and number of iterations (from 1 to 15). The number of iterations relied on a bagging approach that resampled the example set and retrained the model in an iterative way resulting in an aggregated prediction score.

As the number of iterations increases the variance decreases and the Precision increases, however the cost is runtime.

The question thus is how to optimise these parameters, in conjunction with other parameters, to produce a maximally robust, high performant and fast model.

Precision in Link Prediction for masked edges (blue) and human validated edges (red) across differing model parameter settings.

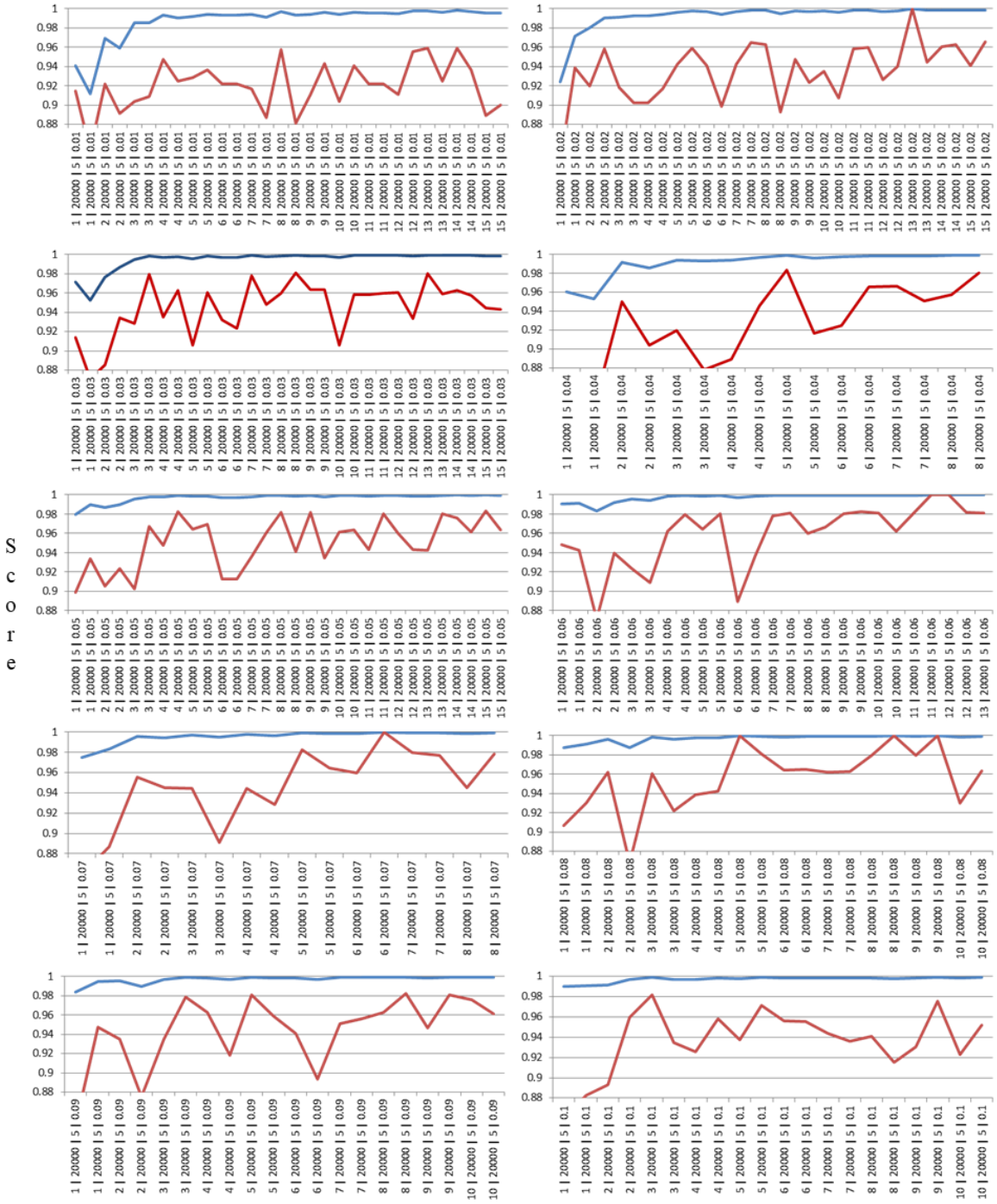


Figure 4.19. A panel of line charts illustrating how the variance of Precision changes as the number of iterations increases, across ten sets of parameters ranging from a proportion of masked edges of 1% through to 10% across the test data.

In response to this question the aggregated variance across a range of iterations was examined (see Figure 4.20. pane 1. and 2.) showing a step improvement in variance up to four iterations, in both masked edges (blue line) and human validation set (red line). At this point variance improvement plateaus, and maximal improvement in variance around the 0.05 masking regime. To place these results in more context the twenty executions on 5 iterations generated masked Precision scores ranging from 0.9923 to 1, and human validated Precision scores of 0.9057 to 1.

Therefore, it is clear from this exploratory analysis that iterating the model and taking a summed probability metric was a performance improvement in terms of both variance and Precision. The decision of how many iterations is optimal is context dependent however based on the initial results the default parameter settings have been set at 4 iterations and 0.05 masking ratio.

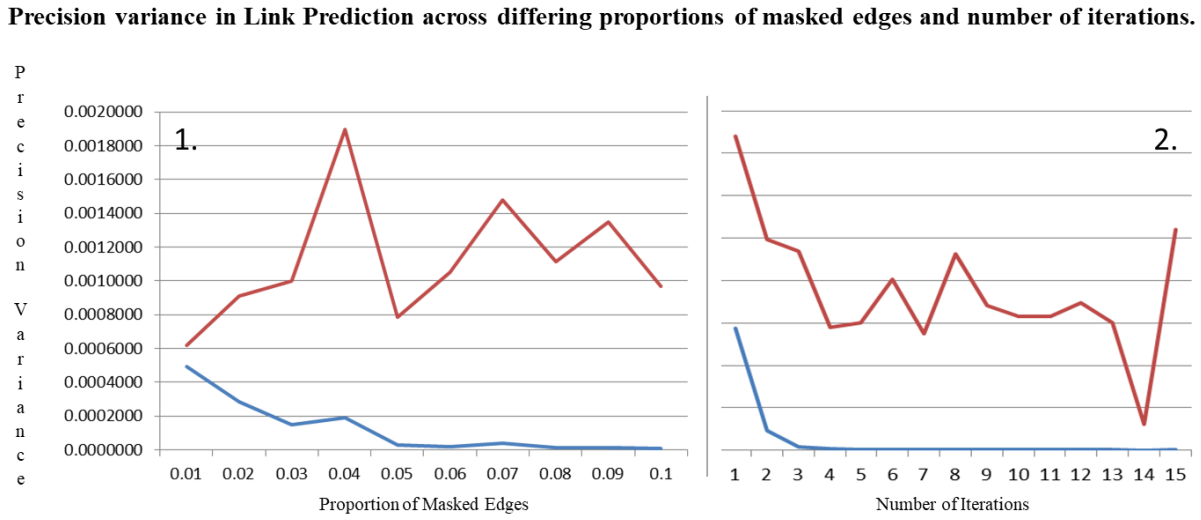


Figure 4.20. A panel of two line charts illustrating how aggregated variance of Precision changes as the proportion of masked edges increases (1.) and as the number of iterations increases (2.) across the test data.

Performance

The datasets which were used to evaluate the LP model include the Sanctions, Dark Network, Offshore Leaks, NZCO, and fused version of all of these datasets. Each dataset was entity resolved with predictions used to contract the graphs. This reflects the conditions of how LP would be deployed in the real-world. Evaluating performance across this heterogeneous set of datasets is useful as it tests the LP model across a range of datasets with varying sizes, topologies, completeness, quality and data models.

The size difference refers not only to the overall number of vertices but also the differing numbers of person entities each graph contains.

The topology of each graph has a significant impact on LP performance. The Sanctions data is close to a set of complete components, exemplified by a very high global transitivity (0.9794). The Dark Network is fundamentally a core peripheral structure, with the giant component comprised of very rich pockets of interconnections. The Offshore Leaks data has very low global transitivity and has a disassortative degree indicating the presence of stars or hubs. The NZCO data has a treelike topology with a high assortative degree, meaning that vertices with similar degree are connected. This means that whilst a significant number of supernodes exist, the predominant topology does not include star like topology.

The completeness also varies across the datasets. Completeness here refers to how closely the number of material edges mimics the real-world, and also how complete vertex attributes (such as date of birth) are. Completeness has a complex relationship with our ability to predict links. In instances where the data is quite specific and focused on discrete clusters of vertices (e.g. the Sanctions data) the global transitivity is high and there is limited opportunity to predict additional links, and these predicted links are of little value (e.g. no weak tie predictions were made on the Sanctions data). However, the NZCO data is an example at the other end of the dimension, as there are initially no person to person relationships included, and so the link inference step is critical to logically derive a coarse foundation of obvious edges, on which then to deploy the more nuanced link prediction model to detect those more latent edges. The opportunity was considerable with the NZCO and so the LP model identified a significant number of inferred links (1,127,002) and predicted links (1,796), with many of these predictions (428) being highly valued weak ties. Completeness can also vary within graphs. For instance, the Dark Network graph is in fact a fusion of many manually annotated datasets derived from investigative intelligence work. Therefore, there will be some areas of that graph that are rich and accurate, and other areas that are sparse, biased and inaccurate. Key vertex attribute data varies across and within graphs. For example, the Offshore Leaks data has no date of birth data, and the NZCO has very little date of birth information available.

The quality of the data is another key element. NZCO data has minimal data validation steps, however Dark Network data has corroborative steps ensuring errors are minimised. For example, within the Offshore Leaks and NZCO data entity types are neither exhaustively provided nor highly accurate – a significant number of entities are simply of an unknown type. And of course, any criminally focused data well undoubtedly contain error as the actors not only withhold true data, they also provide misinformation.

In terms of data models we have already touched on the bipartite nature of the NZCO data, however there are many other subtle differences. A key example is where transactions are represented as nodes rather than edges or hyper edges (a hyper edge is an edge which can include more than one source or target node). This data modelling decision is completely sensible however these decisions have flow on repercussions for LP.

These interdependent factors have a complex interaction on model performance and will each contribute to the natural variation of LP performance.

The model parameters selected for the evaluation model were based on the experimentation from the development phase. Care was taken in ensuring parameters were kept as consistent as possible, except for training set size which was increased in line with the size of the input data.

The runtime across the datasets, relative to the ER runtime, was very quick with the LP model taking just over two hours to run on the fused data deployed in the R language on a Windows 10 environment with a CPU utilising Intel Xeon @ 2.20GHz (8 cores) and 64Gb RAM.

The performance of the LP model cannot be measured in standard ways. It is simply inappropriate to use traditional training and test set metrics in isolation to quantify performance. The reason being that the intent is to predict links that are unknown, and therefore simply masking a subset of known links artificially does not mean we can simply ignore unobserved edges. Indeed, these unobserved edges may be fundamentally different to the edges that are masked, due to a number of reasons including the methodology in data collection. Standard metrics are used for an initial sense check to ensure the model has predictive power and is not over-fit, however the key metric here (along with runtime) is the number of link predictions (predicted edges) that exist in the real-world but remain unobserved in the data. The only way of quantifying this unknown is by taking a sample of predictions and getting an expert to validate whether each predicted edge actually exists in the real-world or not. So, a sample of 100 (or all of the 60 Sanctions predictions) predicted real missing edges was taken from each LP model execution across the datasets and was assessed. An accuracy ratio metric was generated (i.e. how many correct / total predictions), in combination with a breakdown of how many predictions were weak ties and how many were strong ties. The results of the all the performance related metrics are outlined in table 4.10. Each LP model execution was conducted on the entity resolved contracted version of that dataset. The “Fused data” refers to all of the data fused via entity resolution contraction. The fused dataset reflects most closely how the LP model would be deployed in the real-world.

Table 4.10. This table outlines the results of the LP performance on the four evaluation datasets and Fused data.

Evaluation of the Link Prediction model

	Sanctions	Dark Network / STR	Offshore Leaks****	NZ Companies Office	Fused Data
Data					
Vertices	~23,000	~280,000	~1 m	~8 m	~9 m
Edges	~44,000	~900,000	~2m	~90 m	~90 m
Persons	~14,000	~55,000	~300,000	~2 m	~2.5m
Global transitivity pre LI	0.9794	0.0841	0.00058	0.0131	0.0067
Global transitivity post LI	0.9794	0.0871	0.00062	0.0162	0.0077
Model parameters					
Training set size	5,000	10,000	20,000	50,000	50,000
Cost	5	5	5	5	5
Kernel	radial	radial	radial	radial	radial
Iterations	5	5	5	5	5
Percentage edges masked	0.05	0.05	0.05	0.05	0.05
Prediction cut-off [0-1]	0.3202	0.2892	0.5	0.9764	0.6260
Runtime (seconds)	107	266	579	11,430	7,919
LP model performance					
Pairs assessed (test set)	19,885	80,297	31,849	1,538,290	1,002,231
Precision *	1	0.9941	0.9931	0.9945	0.9963
Recall *	0.9995	0.9469	0.9950	0.9907	0.9907
F measure *	0.9997	0.9699	0.9941	0.9926	0.9935
Link Inferences	16	1,813	21,406	1,127,002	742,169
Link Predictions **	60	325	188	1,796	3,094
Accuracy **	0.8	0.8417	0.84	0.83	0.78
Weak ties | Strong ties ***	0 | 60	10 | 315	165 | 23	428 | 1,368	1,233 | 1,861

* Based on the masked and actual edges in the test set.
** Based on sample of predictions via human validation.
*** Weak ties defined as the pair having a graph distance of 3 or more.
**** Offshore data has no date of birth data.

Firstly, we can see that in all instances the test Precision, Recall and F-measure are extremely high which indicates the model is predicting the target set and is not over-fit. However, we are really interested in the number of link predictions (the number of FP's and TN's considered predictions of missing edges that actually exist in the real-world), and how accurate these predictions are.

Interestingly, we get a very consistent set of accuracy scores (0.78 – 0.84) that are accurate enough to demonstrate real-world applicability and a firm basis for further development. Comparison against other LP approaches deployed on criminal datasets under similar conditions creates context to interpret these results. Rhodes and Jones (2009) used a Bayesian inference model to detect unobserved edges in a terrorist group (22 persons) and generated accuracy rates of between 0.26 and 0.45 in the assessment of 136 pairs. Fire, Puzis and Elovici (2013) built a machine learning model using topological features using an open source terrorist dataset (244 vertices and 840 edges). Berlusconi, and co-authors, (2016) built a topologically based approach using common neighbours, Katz index similarity and Structural Perturbation Method (Lu et al., 2015) predicting marginal links removed throughout the investigation/judicial process on an organised crime network (182 vertices and 549 edges). The results in both cases looked interesting however Fire, and co-authors, (2013) used AUC as the sole metric, which while convenient, fails in this context to mimic how the

technology would be deployed in the real-world. Berlusconi, and co-authors, (2016) focus was also not so much link prediction but ranking edges previously discarded for their inferred importance. Additionally, scalability and generalisability are two big unanswered questions in both studies which undermines their contribution to applied settings that fall within the scope of this work.

The distinction between weak and strong tie predictions gives insight in terms of the value of the link inference step and the dependence on that step when attempting to detect the more valued link predictions involving a graph distance of 3 or 4 hops. It is this set of predictions that generate high value. However, it is important to note that weak ties are notoriously difficult to consistently and accurately predict as they depend on the more random and unbounded focal closure mechanism rather than more deterministic bounded cyclic closure.

What kind of relationships does the LP model infer and predict?

The inference sub-module within the LP model currently generates link inference with co-residence and co-office holding. These have been chosen due to their wide applicability and generalisability, however many link inference possibilities exist that are more dependent on the data model and domain.

Analysis of the link predictions has provided insight into subtypes. Familial relationships are a common occurrence, across the differing datasets. The identification of aliases and identity fraud is another common subtype. The LP model utilises a completely different methodology from ER and is not so dependent on vertex attributes, and so goes beyond entity resolution. As such, LP identifies topological similarity between a pair of vertices at a more abstract level and therefore does not rely as heavily on vertex attributes such as name and date of birth to predict similarity.

The identification of several subtypes of LP creates the opportunity to create more nuanced LP models which specifically target missing edges at a more appropriate level of abstraction, generating higher accuracy and speed in tandem with that increase in specificity.

4.2.3 Deployment

The LP model has been developed into a wrapper function written in R and contained within the closed source R package LinkDiscovery. The main wrapper functions arguments consist of:

- Graph: the input graph (default = g)
- Proportion: the proportion of edges to mask (default = 0.05)
- Hops: the number of hops when selecting the example set (default = 3)

- Size: the size of the set used for training the SVM (default = 5,000)
- Link_Inference: whether to conduct link inference (default = TRUE)
- Iterations: the number of model iterations to conduct (default = 5)
- FP_Class_Min: the minimum prediction bound for classifying link predictions (default = 0.25)
- FP_Class_Max: the maximum prediction bound for classifying link predictions (default = 0.75)
- Supernode: the degree threshold at which a node is considered a supernode (default = 2,000)
- FP_Class_Method: the method to use when defining what is a predicted link and what is a TN (default = 0.995; other options include "Betamix")

The output that is generated includes a table with the following:

- Source_ID: The id of the Source node
- Target_ID: The id of the Target node
- Souce_Label: The label of the Source node
- Target_Label: The label of the Target node
- Edge: [0,1] Whether the edge exists in the original data
- Masked_Edge: [0,1] Whether the edge was masked during LP
- Path_Length: The graph distance (shortest path length) between the Source and Target node
- RAI: The RAI for this pair
- JC: The Jaccard Coefficient for this pair
- JW: The Jaro-Winkler distance for this pairs labels
- Normalised_Prediction: [0-1] The normalised summed prediction from all iterations
- SVM_Class: [0,1] The LP class generated from the bagged SVM using a 0.5 threshold
- Missing_Edges: [0,1] The LP class generated from the bagged SVM using a user defined threshold

In addition to this table, a collection of metrics are generated including:

- Runtime,
- Date and time,
- Parameter settings (Cost, Kernel, Training set size, Iterations, percentage of edges masked, missing edges threshold
- Number of pairs assessed (i.e. test set),
- Precision, Recall, and F-measure for the training and test set

- Number of link inferences, number of link predictions, number of LP weak ties, number of LP strong ties,
- Pre and Post LI global transitivity.

The last object generated is the original input graph with the LI edges added.

In terms of real-world deployment it is logical to enhance the quality of the data as much as possible through ER and provide a contracted entity resolved graph as the input to the LP model. The LP model can then generate a set of predictions and inferences in the form of triples that can then be used in combination with the ER prediction triples to generate a fused graph with predictions presented as edges (e.g. red for ER and blue for LP). In this way the user can determine how best to use the predictions for optimum value. For example, a subgraph can be presented of a fused criminal subgraph with only high certainty predictions materialised, and the user can then incrementally add relevant predictions to augment the subgraph and generate leads for further investigation or intelligence collection.

4.2.4 Conclusion

Data completeness within the criminal domain is a real problem. One element to this completeness is the absence of relationships in the data that exist in the real-world – missing links. Link prediction technology has been developed to address this missing link problem, however the deployment of this published technology to real-world applications has limited utility. This limited utility is based on a lack of scalability, generalisability and accuracy. The LP model developed here is designed to improve on this situation efficiently detect missing links, when given a social network, with an accuracy ~ 0.8 (dependent on the strategy used to materialise the link predictions), predicting both strong and weak ties to enable a more valuable application in the criminal domain.

The evaluation of the developed LP model's performance is encouraging with the model successfully predicting a significant number of links across each dataset with a consistent accuracy of around 0.8. Of these predicted links a significant proportion are the highly valued weak ties that generate latent knowledge. The computational efficiency and scalability of the model indicates utility in graphs up to a size of ~ 9 million nodes and ~ 90 million edges with a runtime of around two hours, performance that is far from prohibitive for many criminal domain applications. However, this model has been developed in a way that enables it to be run over a distributed computing platform, greatly extend its scalable applicability. There are a few LP models applied to the criminal domain (Rhodes & Jones, 2009; Fire et al., 2013; Berlusconi et al., 2016) which provide some basis for comparison, however scalability and generalisability of these approaches simply remain untested and unknown.

The LP model has been developed into a wrapper function written in R and contained within the closed source R package LinkDiscovery.

4.3 Summary of "Make Data Exploitable"

Entity resolution and link prediction are two critical elements to any advanced intelligence approach within the criminal domain, and even more so if the goal is to identify those entities that bind and maintain the structural fabric of criminality. The two modules presented outline applied yet generic components to make data exploitable.

The entity resolution model is a semi-supervised learning approach that iteratively identifies obvious and potential non-obvious pairs, collecting explicit and engineered implicit features through the deployment of novel technology such as the Proper Name Classifier, Proper Name Origin Classifier, Reference Graph Algorithm (Robinson, 2016), and Collective Entity Resolution. This feature extraction process utilises the iterating input data, an onomastic gazetteer, and prediction data. The set of pairs and features accrued through this process represents pairs believed to be duplicates – known as validated – and those pairs that have significant equivalence but are not believed to be duplicates – known as invalidated. This set of pairwise features are the input to a Support Vector Machine that generates probabilistic entity resolution predictions. The evaluation on real-world data has yielded encouraging results, in accuracy, generalisability, runtime, and scalability, outperforming commercial software.

The link prediction model adopts a machine learning framework (utilising RPART or SVM), using a number of features that concentrate on homophily at both the micro level (e.g. comparing pairwise degree) and meso level (e.g. assortativity of proper name origin), and graph distance measures (e.g. RAI) to give a set of input variables that measure the different theoretical underpinning sub-elements of cyclic closure, focal closure, and preferential attachment. Link inference is a key element that augments the data, before the masking of observed edges (default of 5%) is undertaken, model training and model deployment is undertaken. This process of edge masking, model training, and model deployment is undertaken in a loop (default of 4), generating an ensemble set of model output that is combined and used to generate a robust set of probabilistic link predictions.

The evaluation of both the ER and LP modules on real-world data yields results with high utility, beyond other commercially available software. Improving the quality of data is a critical step to create a firm basis for applying knowledge discovery approaches, which we will now turn to.

Discover Knowledge [chapter 5]

The computational work done to this point is aimed to generate the best quality data possibly. That means attempting to reconstruct the data to as closely as possible represent the real-world, explicitly recognising the incompleteness of the data. Having generated a dataset that is both rich and accurate creates the opportunity to discover meaningful latent knowledge. By latent we mean knowledge that is not patently obvious or merely an aggregation of micro events. Of course the data is now in a state that can also be used for query and ego based search and transformed and augmented to provide a basis for ad hoc exploratory analysis, however much can be done computationally to provide a more advanced proactive capability.

To generate these proactive insights generic approaches can be allied to contextual domain knowledge that together can uncover knowledge from multiple perspectives. These perspectives range from the micro through to the meso and macro perspectives. The micro perspective is squarely on the individual - where the goal may be the identification and prioritisation of an entities risk (e.g. the risk an entity poses of trafficking illicit drugs). The meso perspective focuses on functional groups of entities – where the goal may go beyond the identification and prioritisation of groups into understanding the context of how groups inter-relate and functionally operate. A meso goal may, for instance, be uncovering the weaknesses of a group and constructing a mitigation strategy based on this knowledge. The macro perspective (commonly referred to as strategic) focuses on the entire system or network with the goal of trying to understand how the system operates so strategic resource deployment and strategy can be best executed. A macro goal may, for instance, be the contextual categorisation of all functional groups across a system (e.g. the illicit drug supply chain) so resource can be best targeted to dampening that system and achieve the maximum enduring impact.

The key concepts of graph partitioning and contextualisation will now be covered, which in conjunction create contextual meta-data, before we examine the "GraphExtract" algorithm which is deployed to discover contextual latent knowledge from multiple perspectives across the micro – meso – macro spectrum (see Figure 5.1.).

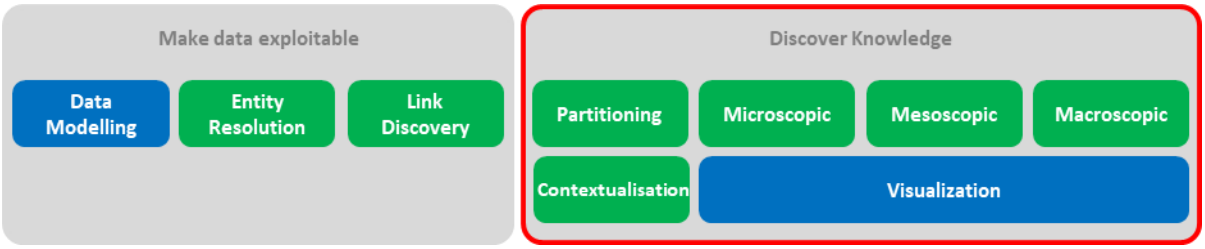


Figure 5.1. This figure outlines the modular design of GCND, with the current focus on the Discover Knowledge section.

5.1 Partitioning module

The partitioning of graphs is a broad cross-discipline focus for research. Its applicability is broad and remains a key element in the study of social networks. Graph partitioning is the separation of nodes into distinct groups based on their connections. Partitioning comes in many flavours. Partitioning based on components is a straightforward approach to extract knowledge from the graph, however where the graph contains large complex components the value of this approach diminishes and alternate approaches are needed. Community detection is a key area of study that, although no standard definition exists, has generated significant applied value. The goal is to generically identify non-overlapping (partitions) or overlapping communities that contain "significantly" more intra-community than inter-community relationships. A number of algorithms have been developed to computationally efficiently identify communities. The point of identifying communities based purely on their connections gives us the opportunity to identify functional groups of entities that coalesce. This then gives insight into what groups of entities are working together. Of course, the knowledge that can be generated through community detection in this context is heavily dependent on the quality, completeness and the conceptual basis of how the graph is constructed. The conceptual basis of graph construction refers to only including relevant data. This decision however is theoretical as we often do not have knowledge of exactly who forms each functional group. So, there are important decisions to be made, dependent on the domain context, about the meaning or semantics of edges and nodes. For instance, identifying a functional group of criminals from Facebook data alone would be fraught with difficulty, whereas constructing a graph from co-offending data, the Offshore Leaks, and phone data may provide a firm basis from which to detect a community of entities that form a functional criminal group. It is important at this point to delineate between a functional group and a non-functional group. Organised criminals may overtly identify themselves (e.g. outlaw motorcycle gangs) or may be more covert in their membership to a formal group. What is important to acknowledge here is that formal organised criminal groups often do not orchestrate criminal acts in a way that mimics legitimate hierarchical business structures. It is much more common for autonomy to pervade with members empowered to engage in their own niche activity, whether legitimate, illegitimate or a mix of both, dependent on the opportunities that arise. The formal group creates reputational backing, network opportunities for resource and imposes a standard of behaviour that allows the group a sustainable existence, and enables members to flourish in their activity. Functional groups however refer to fluid groups that have a functional purpose that may or may not have members formally linked to organised crime groups, or may rather be entrepreneurial in nature. Either way it is these functional groups, in the context of the umbrella organised criminal groups, that are of interest.

So, a key goal is identifying functional and formal groups within the system. Doing this generates context of who is coalescing and unlocks the opportunity to view the system from meso and macro perspectives.

Three partitioning approaches are used within GCND; component detection (assuming the graph is undirected), the Louvain community detection algorithm (Blondel et al., 2008) and the "GraphExtract" algorithm (Robinson & Scogings, 2018). But before we explicitly attempt to identify functional criminal groups, it is important to generate applied context.

5.2 Contextualisation module

A range of graph theory metrics are generated within GCND and used in a myriad of ways, including; degree centrality, local transitivity, betweenness, brokerage, diameter, global transitivity, and degree assortativity (see literature survey for details). These metrics are incredibly useful when used in tandem with contextual metrics and analysed by domain experts. Contextual metrics for example could include regular expression-based approaches to tag vertices based on the corporate entity type they represent (e.g. trust, limited partnership, limited company, anstalt, societe anonyme), categorise suspicious transactions as per a money laundering typology, and using address parsing routines to identify jurisdictions entities have ties too. The use of graph theory / SNA metrics in the criminal domain is not new so let's focus on two extremely useful novel metrics that are complex systems focused, and are deployed within GCND – supply chain metric and attitude.

5.2.1 Supply chain inference and identifying "Super-brokers"

Organised criminality is about money. Of course, there is more to it than just money but wealth generation is the core driving concept underpinning organised crime across the globe. A closer examination of wealth generation uncovers a set of roles, skills, resources and relationships that are required to generate said wealth. A useful perspective to provide context is that of the illicit supply chain (see Figure 5.2.).

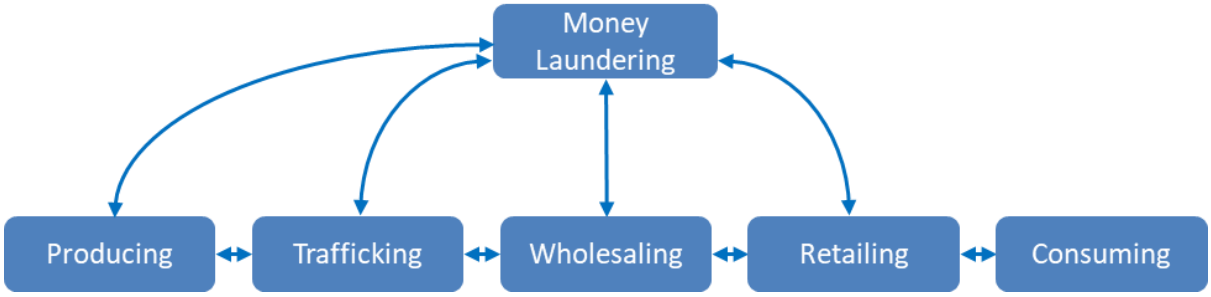


Figure 5.2. This figure provides a basic mapping of the illicit drug supply chain.

Mapping graphs of entities to the illicit supply chain creates the necessary context enabling a systems view. This systems view then creates the opportunity to uncover latent knowledge across micro, meso and macro perspectives. For example, at the micro perspective we now have enough context to go beyond the inference of brokerage roles, based purely on coarse (i.e. all edge types) or even fine-grained (e.g. co-offender) edge type filtering. Having identified where in the supply chain an entity focuses their attention we can then identify whether they have a direct relationship with an entity that engages in an adjacent phase. If the target entity (e.g. a trafficker) does not have a direct relationship with someone that engages in an adjacent phase (e.g. a wholesaler) either there is another relationship with a wholesaler that we do not have visibility of or there is an indirect relationship with a wholesaler through a third party – a broker – who does have a relationship with a wholesaler (Natarajan, 2006). Logically this is likely to occur in instances of where trafficking is undertaken by offshore nationals who have reduced contacts with the domestic wholesale market. Ethnic based brokers that have contacts with ethnic enclaves and domestic drug markets are perfectly positioned to facilitate the flow of commodity from one phase to the next. Conducting this analysis across all entities that are observably involved in the illicit drug supply chain we can not only identify those entities that are more likely to perform brokering roles, we can also identify, with increased confidence, those brokers that are significantly over-represented. It appears a very small set of brokers indirectly connect a large proportion of entities engaging in the supply chain. This small set of brokers, dubbed "Super-brokers", is significant in terms of the criminal system functioning efficiently and effectively. Therefore, intervention targeting these key system vulnerabilities may prove considerably more successful in inhibiting the success of the criminal system than targeting the supply chain in a random fashion. Figure 5.3 illustrates those entities directly or indirectly in the supply chain across the fused data, clearly indicating the significance of "Super-brokers" role in connecting actors involved in the supply chain.

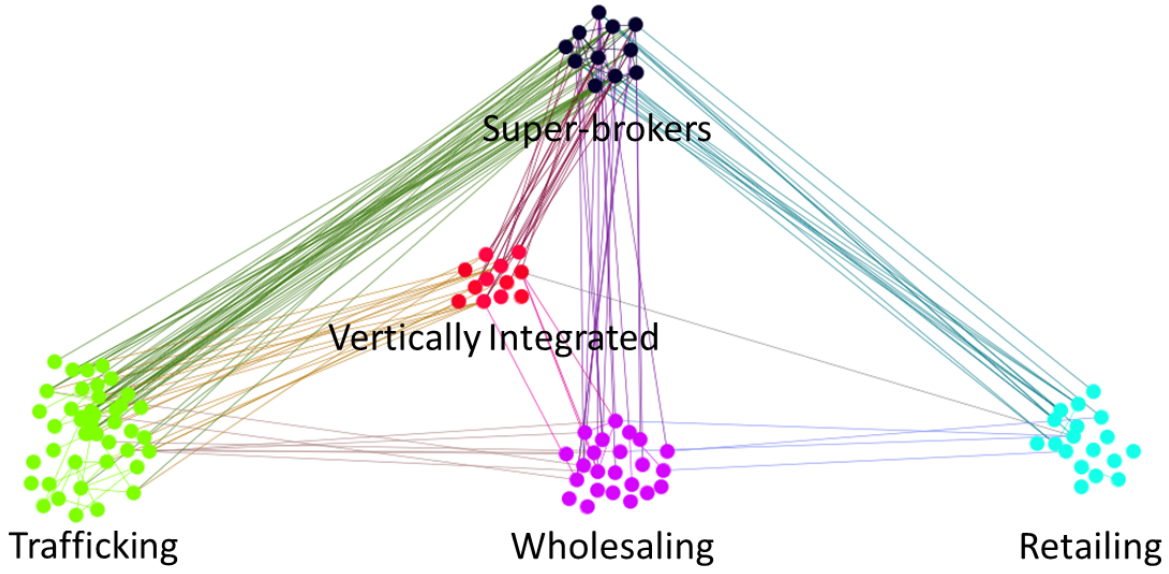


Figure 5.3. This figure illustrates those entities directly or indirectly in the supply chain.

Deployment of supply chain roles

The identification of entities involvement in the supply chain is based solely on a series of regular expressions "tagging" each node with a supply chain role if the text associated to each node contains a specific regular expression. For example, any entity with a reference to "suppl", "wholesal" or "distribut" and a synonym for illicit drug (e.g. "meth", "cocain", "opium"…) is tagged with "Wholesaler". Note that there will be many instances of where a node does not have a role in the supply chain due to either a lack of data or the fact they are not involved in the supply chain.

Subsequent to the tagging of every node for their role in the supply chain, we simply count the number of times a broker connects a source node (e.g. Trafficker) with a node from the adjacent supply chain phase (e.g. Wholesaling) where the source node does not have a direct relationship to a node in the adjacent phase.

From a micro perspective it is now possible to identify those key "Super-brokers" in the network and make a contextual decision on which to target for intelligence or intervention resource. From a meso perspective we can also measure what groups use the services of these "Super-brokers" and not only infer the paths these groups acquire product through the supply chain but also use this as an element to measure the sophistication of groups that use brokers to not only source commodity more efficiently but also perhaps insulating themselves from perceived risk (e.g. law enforcement). From a macro perspective we can also use "Super-brokers" as a key feature to measure how the criminal system is changing over time, and how to adapt intelligence and intervention strategies. For example, in response to an intervention strategy focusing on "Super-brokers" there may be an increasing

perception of risk associated to that specific role and so perhaps a foreseeable response would be for wholesalers to limit their exposure to brokers and spend more time building relationships directly with traffickers and or groups becoming more vertically integrated to limit this risk.

In terms of performance the value of the notion of “Super-broker” in an applied sense is yet to be comprehensively measured, although the metric undoubtedly contributes to intelligence and investigations by providing an objective metric with high face validity. Runtime across the fused data set is ~ 27 seconds.

5.2.2 Predicting Attitude

From a behavioural perspective the prediction of the attitude of entities within the criminal domain is a very important element. This is because the attitude of persons, in combination with other concepts such as opportunity and access to resource, can then start to form the basis to measure the risk associated to persons, groups, sub-systems and the entire system. Risk metrics can then be utilised appropriately to inform decision-making.

However, the incompleteness of the data means that generally only a fraction of entities will have requisite data to enable an accurate prediction. This is where propagation or diffusion methods can be used to impute attitude scores mimicking the flow of information through a network. The applied basis of this idea has been derived from the empirical analogues that were demonstrated in 2007 by Christakis and Fowler who showed diffusion patterns – which they refer to as hyperdyadic diffusion – in obesity, smoking and alcohol consumption.

Construction of attitude propagation

The technology used to deploy the idea of attitude propagation is a radial walk algorithm that enables parallel duplication (Borgatti, 2005) – mimicking the diffusion of multiple information ‘packets’ throughout a graph simultaneously. Eigenvector centrality is the basis for the mechanics of the radial walk. We will refer to the model as Radial Walk Attribute Propagation (RWAP).

Eigenvector centrality uses radial walks to measure how influential a node is. The score is derived by iteratively summing each node’s neighbours scores (beginning with each node neighbour equal to 1) and then generating a proportional score by dividing this score by the largest value in the graph, until the scores reach equilibrium. In this way "high-scoring" neighbours contribute more to a node’s score than "low-scoring" neighbours. Eigenvector centrality is designed to identify those entities that are connected to highly connected or prominent, entities, which can be interpreted as measuring status. So, in the same way RWAP

should have predictive power assuming strong homophily is present within the criminal fraternity.

The method to generate the observed scores is to identify those persons within the graph that have a specific conviction or allegation and allocate a score [0-1] based on the normalised ranked average custodial sentence for that offence. Custodial statistics for New Zealand were sourced from NZ Statistics public website and each aggregate offence type was ranked and scored with the most serious offence, murder, scoring 1.

This aggregated offence ranking was then used in combination with regular expressions to detect which persons within the data had what convictions or allegations and were ascribed the appropriate score. In instances where an entity had more than one offence the most serious would be used for the score. Of course, this method could have been more nuanced taking on additional dimensions such as a decay factor for the time since the conviction/allegation, acknowledging that a person's attitude can change significantly from being an 18 year old drug dealer to a 40 year old father and accountant. However, this model was only ever planned to be an initial coarse representation of attitude and avoid creating dependencies on specific data attributes such as date and edge type. Additionally, at this point in model development there is significant value in creating a simple model that has fewer moving parts enabling less complex exploration of the results and a firmer path to subsequent extensions.

The second phase of the model uses a radial walk propagation algorithm (based on eigenvector centrality) using the attitude scores as the observed seed scores to propagate through the network iteratively via radial walks.

The absence of observed scores – let's call these unobserved scores – has a practical implication. Firstly, as observed scores are used as the seed scores for the algorithm how do we seed those vertices with an unobserved score? Secondly, can we utilise the data beyond a local sense more effectively to remedy the severe lack of observed scores within the criminal domain (e.g. the Dark Network has ~ 1% of observed scores). Two solutions have been proposed to remedy the lack of data. Version 1 of RWAP (RWAP1) utilises a fixed seed in line with what kind of mean score we would find in the general population (default of 0.1). Version 2 of RWAP (RWAP2) utilises a more nuanced approach taking into consideration the observed scores from a mesoscopic perspective. Using the data beyond the local micro context not only makes intuitive sense but is also aligned with the notion of emergent properties such as the collective influence on individuals (e.g. group polarization; Myers & Lamm, 1976). This idea was applied by executing the Louvain algorithm to detect communities and generate the mean observed scores for each community and apply this mean score to all relevant vertices. The

maximum observed scores were experimented with but generated inferior results to the version of using the mean.

The parameters for the algorithm include how much a person's attitude score should rely on their own base attitude (parameter *i*) versus the attitude of their neighbours (parameter *j*). The parameter *iter* refers to the number of iterations to execute towards convergence before termination. The parameter *Q* refers to the number of neighbours ordered observed scores to consider when summing.

Performance measures

The performance of the model was determined by leave one out analysis. This is based on masking the observed score of one node prior to execution and measuring the algorithm's predictive output for that masked node. For optimisation purposes this leave one out routine was run iteratively on 642 different nodes that have observed scores, within the Dark Network. Conducting analysis in this iterative way, whilst computationally expensive, ensures that the impact of losing any more data is absolutely restricted to the single node under examination at any time. This methodology creates a firm basis for measuring model success as now we have maximised the number of instances from which we can measure performance yet minimising the loss of information.

The amount of information available is not only important in the sense of the number of observed scores versus unobserved scores. The distance between a node and other nodes with observed scores is critical. This distance (or reachability) from information is a key element to explicitly explore, as it is self-evident that it is easier to predict a node's unobserved score when they have neighbours that contain observed scores contrary to instances where the closet node that has an observed score has a graph distance beyond one. Figure 5.4 displays this phenomenon concisely illustrating the association between observed scores (x axis) and predicted scores of masked vertices (y axis) using the leave one out methodology, across nodes who have neighbours that have observed scores (N_1), nodes that have 2nd order neighbours that have observed scores (N_2), nodes that have 3rd order neighbours that have observed scores (N_3), and lastly nodes where the closest node with an observed score is beyond 3rd order neighbours or there is simply no reachable nodes with an observed score (N_{4+}).

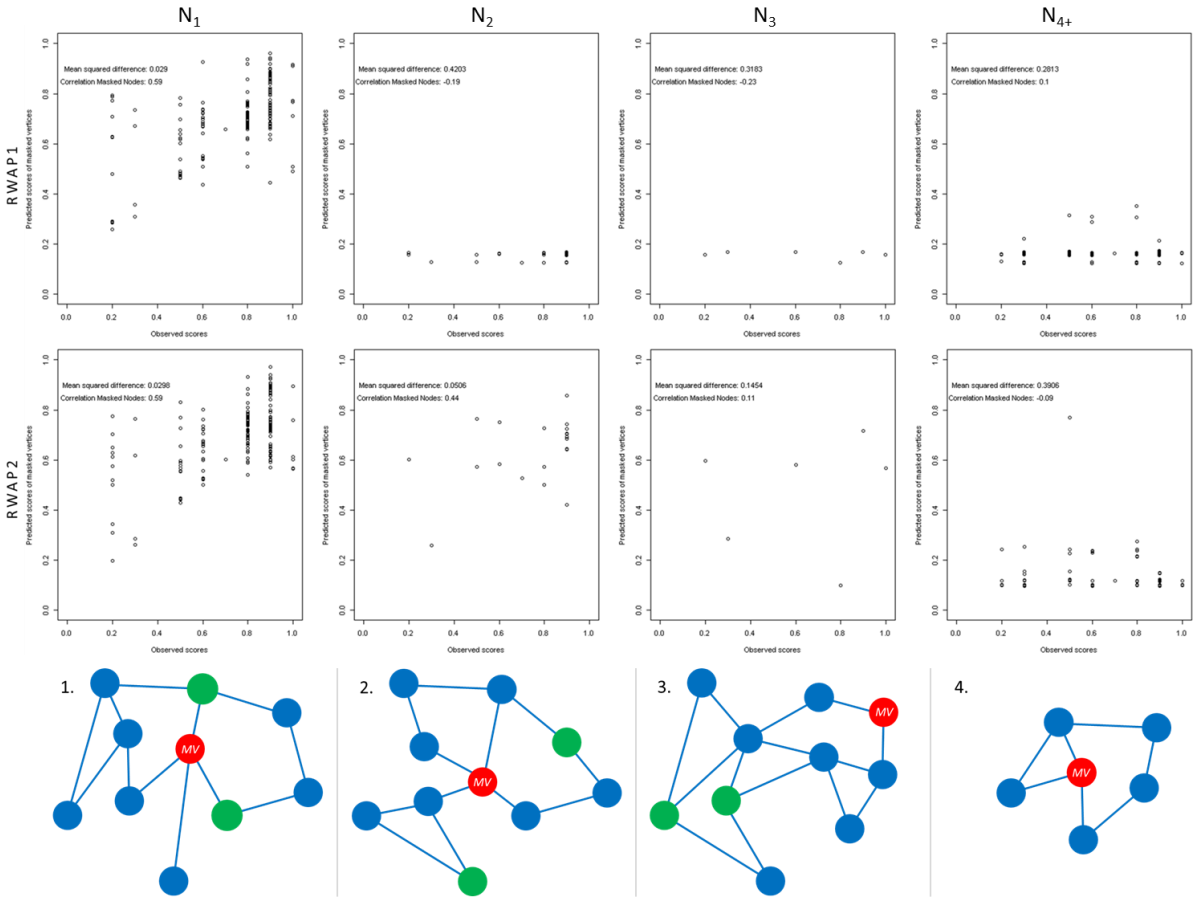


Figure 5.4. This figure illustrates the performance of RWAP1 in the top row and RWAP2 in the middle row, with examples of how each column represents differing sub-groups based on their reachability to nodes with an observed score.

The key metrics used to measure model performance across these four subsets of reachability are squared difference between the observed score and the masked-predicted score using leave one-out, and correlation. We can use these two metrics to optimise the parameters (highest correlation and lowest squared difference) and generate an optimal model in terms of predicting unobserved scores. Or in other words, a model that gives us the best predictive ability to impute unlabelled nodes. For a sense check we also record the overall correlation of predicted scores with observed scores. Using the mean squared difference (MSD) to optimise the model has its limitations and bias, however it is conceptually intuitive and straightforward. The core bias is that using mean squared difference optimises models that reduce extreme variation in scores, so a model that makes ten predictions of 0.4 where the masked vertices scores were all 0.1 would score a MSD of 0.09 and a mean difference of 0.3 in comparison to a model that makes eight predictions of 0.25 and two of 1 where the masked vertices scores were all 0.1 would score a MSD of 0.18 and a mean difference of 0.3. So, using MSD rewards models that minimise extreme scoring error. Correlation was used as a secondary metric to gauge performance success. Using squared difference and correlation, in combination with the visualisation displayed in figure 5.4 enables a clear basis to understand RWAP performance.

The difference in performance between RWAP1 and RWAP2 is noticeable. The correlation across all observed scores without masking is 0.86 and 0.94 for RWAP1 and RWAP2 respectively. Examination of the leave one out masking indicates that RWAP1 has predictive power in the group of masked vertices that had at least one neighbour that had an observed score (column N_1) with a correlation of 0.59 and mean square difference of 0.029. However, this performance drops off to a correlation of effectively zero, given the small sample size, in instances where the closest observed score is beyond a node's immediate neighbourhood (see Figure 5.4.). The inference can be drawn that using the radial walk approach in instances with very limited information available – only 1% of nodes have an observed score – generates functional predictive value locally rather than in the more global sense across the network.

In response to this, RWAP2 was designed to extract more global value from the limited information available. This design was based on utilising the local information at a mesoscopic level, or, in other words, using the notion of community and the underpinning social psychology theory and empirical evidence to infer community members seed scores based on other community members observed scores. This was applied by using the mean observed score of each community as the starting seed for nodes with unobserved scores. Using the information in this way retains performance when a masked vertex has neighbours that have observed scores (correlation increases to 0.59 and summed square difference of 0.0298), but it significantly boosts performance in situations where observed scores are only reachable through a graph distance of two (column N_2) and even three (column N_3) - (correlation increases to 0.44 and 0.11 and mean square difference of 0.0506 and 0.1454 respectively). There is no tangible impact on masked vertices that have either no information to predict a score (see example in column N_{4+}) or where there is a graph distance between the masked vertex and its closest observed score beyond a graph distance of three – both RWAP1 and RWAP2 fail to garner any material predictive power.

So, RWAP2 performs somewhat as expected, given prior behavioural diffusion empirical evidence (Christakis & Fowler, 2007), with a reduced but significant predictive performance decreasing in line with the drop in local information availability (see Figure 5.4.). The number of observed scores (642) makes up around 1.16% of the ~55,000 person nodes within the Dark Network graph. Interestingly, the proportion of nodes falling into each of the four subgroups of reachability – direct neighbours (N_1); indirect neighbours with a graph distance of 2 (N_2); indirect neighbours with a graph distance of 3 (N_3); and indirect neighbours with a graph distance of beyond 3 or nodes not reachable (N_{4+}) – is 3.3%, 6.1%, 6.9%, and 93.1% respectively. The obvious important point here is that with a set of 1.16% of possible observed scores we have the ability to predict attitude, with some confidence, up to approximately 6% of persons in the data. This gives some early indications, topologically dependent, of how large the observed score set needs to be to get requisite coverage of attitude predictions across

the entire graph. The second point here is that information collection can be targeted at the most topologically relevant vertices (e.g. high degree and high betweenness) to generate most value out of the observed scores that are available.

Pane 1 of Figure 5.5 illustrates the observed scores in a component of the Dark Network, with pane 2 illustrating the subsequent RWAP2 predicted scores for all nodes. Nodes represent people and the colour represents the attitude (whether observed or predicted) with red equivalent to a score of 1 – the most non-compliant attitude –through to light orange which is equivalent to a low score (e.g. 0.1) – a minimally non-compliant attitude – with grey representing unobserved scores.

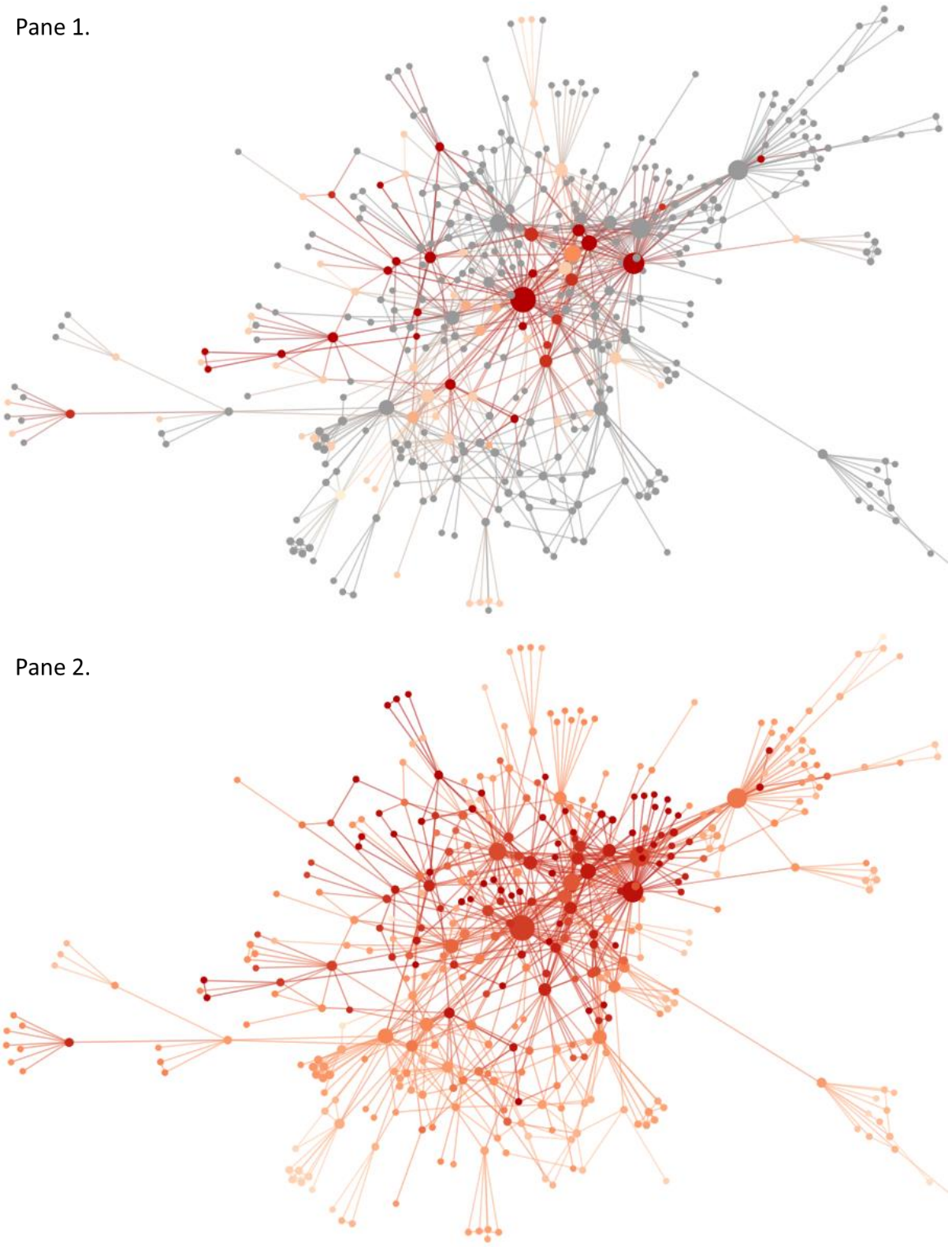


Figure 5.5. This figure illustrates the RWAP2 output on a component from the Dark Network in pane 1, and the propagated version in pane 2.

Generalisability

I have previously demonstrated that RWAP2 has predictive power in instances where persons are directly associated to someone with an observed attitude score. Furthermore, this predictive power has also been shown to decay as the relationship becomes increasingly indirect and distant. But, how does the performance of RWAP2 generalise to other data and other mechanisms of generating observed scores?

The only relevant dataset available to test the generalisability of RWAP2 was the Fused dataset. This dataset tests the scalability of the RWAP2 algorithm and topological generalisability as the topology has changed. The Dark Network is a subset of the Fused Data and contributes 642 of the 1,742 observed scores. Figure 5.6 illustrates the performance of RWAP2 on the Fused Data in comparison to the Dark Network. We can see that the overall correlation between initial observed scores and masked vertices (via leave one out method) declines slightly from 0.94 to 0.90, but the correlation and mean squared difference for N_1 slightly declines from 0.0298 to 0.0271 with an associated increase in correlation from 0.59 to 0.78. The results for N_2, N_3 and N_{4+} for mean squared difference are 0.0815, 0.1817, and 0.5209 and a correlation of 0.14, -0.08 and -0.06, respectively. This indicates a performance decline in terms of indirect connectivity to vertices with observed scores, however visual examination of figure 5.6 indicates very similar patterns to the Dark Network results. From this we can surmise that the small sample size in N_2 and N_3 leaves the results susceptible to significant variance with the addition or deletion of one or two data points. So, any conclusions taken from this evidence has to be couched in this context.

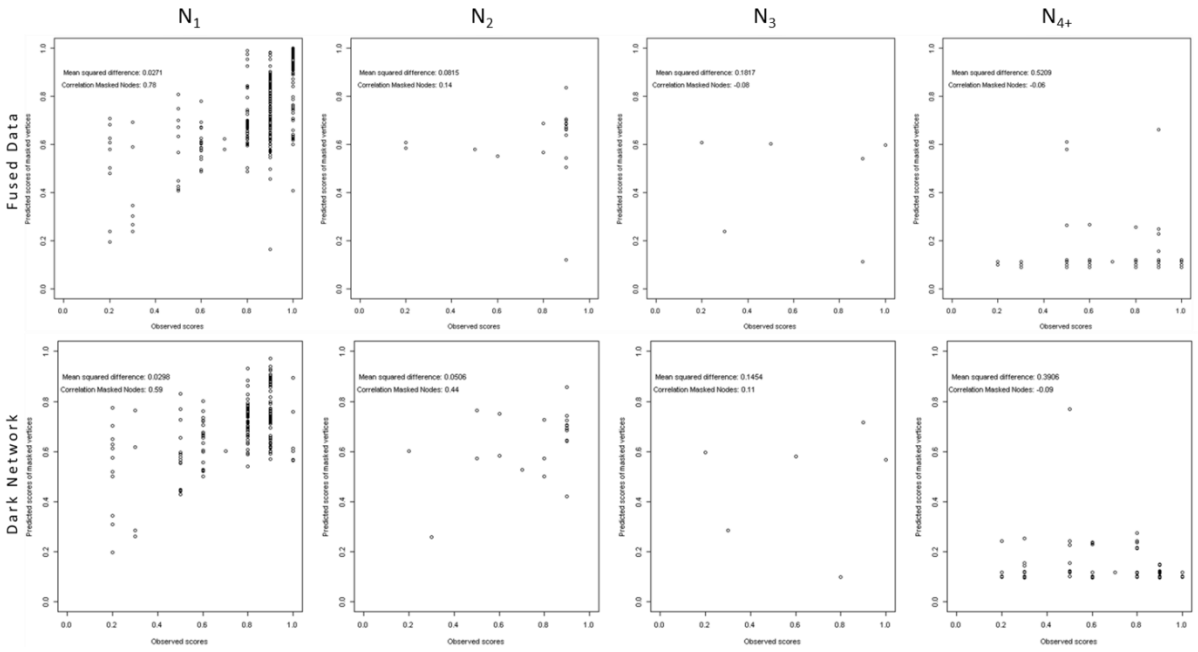


Figure 5.6. This figure contrasts the performance of RWAP2 on the Dark Network in the top row and the Fused Data in the bottom row, across differing states of reachability (N_1, N_2, N_3, N_{4+}).

Table 5.1 summarises the relevant performance metrics of RWAP2 on the Dark Network and the Fused Data. The predictive performance of RWAP2 on direct neighbours is clear, however the performance of RWAP2 on indirect neighbours is less clear, with any firm conclusions not able to made due to the small sample size. There is, however, still some predictive ability in N_2 and N_3 when compared to N_{4+} across both datasets, it is just not clear the extent, and nor is it clear how topology, data quantity and data quality impacts on performance.

Table 5.1. This table outlines the results of the RWAP2 performance on the Dark Network and Fused Data.

Evaluation of Radial Walk Attribute Propagation

	Dark Network / STR	Fused Data
Data		
Vertices	~280,000	~9 m
Edges	~900,000	~90 m
Persons	~55,000	~2.5m
Global transitivity post LI	0.0871	0.0077
Observations	642 | 1.16%	1,742 | 0.07%
Vertices reachability to observed score		
N_{1*}	3.3%	0.12%
N_2	6.1%	0.16%
N_3	6.9%	0.17%
N_{4+} or not reachable	93.1%	99.83%
Model parameters		
i | *j* | *iter* | *Q*	3 | 7 | 3 | 1	3 | 7 | 3 | 1
Runtime (seconds)	3	58
RWAP performance		
Correlation: observed v predicted scores	0.94	0.90
Masked vertices [MD** | MSD*** | correlation]		
N_1	0.1726 | 0.0298 | 0.59	0.1650 | 0.0271 | 0.78
N_2	0.2249 | 0.0506 | 0.44	0.2855 | 0.0815 | 0.14
N_3	0.3813 | 0.1454 | 0.11	0.4263 | 0.1817 | -0.08
N_{4+} or not reachable	0.6250 | 0.3906 | -0.09	0.7273 | 0.5209 | -0.06

* N_n = Neighbourhood$_{[order\ n]}$
** Mean difference
*** Mean squared difference

Another key point to highlight is that the way the observed scores were generated provide a highly unbalanced set of information from which to generate a model. When attempting to predict the attitude of persons within a criminal centric dataset this is not such a problem, however when applying this same approach to a dataset that represents the general public as a whole the observed scores are not such a representative view of the whole. The reality is that the vast majority of persons

attitudes across the general public will be very low, and so as the observed scores from which the masked vertices are derived are based on the presence of convictions imbalance will result. The result of this imbalance is that the performance metrics become less useful in measuring the success of the model as the reachability decreases through N_2, N_3, and N_{4+}. This is because these are the exact areas of the graph where attitudes are more likely to be compliant, and yet we are using a set of masked vertices with disproportional non-compliant attitudes as the measuring stick. The conservative nature of RWAP2, as demonstrated in figures 5.4, 5.5 and 5.6, is that if there is little information available to make a prediction then that prediction will generally be close to 0. This is an important element to the applied model so the instances of predicting non-compliant attitudes in actual compliant persons is absolutely minimised.

Of interest is that the optimal algorithm parameter settings consistently show that a *Q* parameter of 1 is optimal. This means that the algorithm performs most optimally when it only takes into consideration the score of the highest neighbour, ignoring the remainder of neighbours observations. An interpretation of this is that the attitude of a neighbourhood has a strong relationship to the person with the highest attitude score, highlighting potential emergent properties such as group polarisation (Myers & Lamm, 1976). The significant performance improvement through N_2 and N_3 through the introduction of using the mean community observed scores lends further evidence of the influence of the collective on the individual, beyond direct relationships. Assuming this phenomenon is generalised, what are the mechanics driving this? Is it homophily in action where people select relationships based on similar attitudes, and / or is it a case of influence? This of course needs to be couched in the context of the high uncertainty of the data, as we have neither the complete picture when it comes to vertices nor relationships, and the size of data would ideally be extended. Furthermore, the precise definition and measurement of the behavioural element – attitude – under consideration requires extension and corroboration. When these things have been improved a more rigorous approach can be instigated. This however is well out of scope for this paper. Nonetheless these are possible extensions that would enable the development of enhanced iterations of this computational approach and the broader understanding of criminal networks.

Interestingly the same set of parameters were optimal in both conditions, hinting towards the generalisability of the model across variable conditions, and creating the opportunity to test a number of hypotheses in relation to the mechanisms explaining and underpinning criminal networks.

Importantly, from an applied perspective, the runtime indicates a highly scalable approach, that with engineering can be further optimised.

Deployment of predicted attitude in real-world settings

The predicted attitude scores can serve as a building block of assessing risk, which, in combination with other risk elements such as opportunity and access to resource, can form the basis of prioritising intelligence and/or intervention resource. Opportunity, for instance, within the border security domain, could derive from the fact an entity is connected to an import/export business that imports from a high-risk source country, or within the tax domain an entity could own a cash business like a restaurant. Attitude can also give insight at the meso and macro level. From a meso perspective attitude metrics in combination with community detection and supply chain models can give insight into how functional communities of entities are interacting given the context of the supply chain. Groups may be identified that have a "low" scoring collective attitude, which can be interpreted as law enforcement having low visibility of their activity, ranging through to "high" collective attitude where entities have a significant history with the judicial system. At the meso layer coarse inferences can be made which can direct intelligence resource. For instance, if it is observed that a "low" attitude domestic group has newly developed links to a broker or "Super-broker" that is loosely tied to a transnational group that has an involvement in trafficking illicit drugs then it may be decided that intelligence and or investigative resource should be deployed to gather more information so a more accurate risk assessment can be made. At the macro level attitude models can contribute to the measurement of which functional and formal groups pose the highest risk, and importantly give insight into how the whole system is functioning, identifying system vulnerabilities. From a temporal perspective it would be useful to gain insight and understanding of how networks evolve in relation to attitude.

When the data collected is of sufficient quality much more nuanced models can be developed that identify leaders, influencers, and emergent properties of and within groups, using a foundation of social psychology (e.g. balance theory, deindividuation and group polarisation) and psychopathology empirically tested theory. These more complex concepts can be used as a foundation to build models that more accurately predict associated entities attitude and influence on others over time, given the group and neighbours context. However, these more complex models are dependent on high quality data, which places the onus firmly on good data collection, conceptual modelling, data representation, data cleansing, and entity resolution – the foundations of any advanced analytical endeavour.

5.2.3 Conclusion

Contextualisation is a critical step in generating a set of building block metrics that create sufficient domain context. Contextualisation is focused on utilising the explicit data available (i.e. the entities, relationships and attributes) in conjunction with implicit metadata generated through domain inference (e.g. inferring co-residence relationships, inferring corporate entity type) and generic computational metrics (e.g. community detection, degree centrality, local transitivity, betweenness,

brokerage, diameter, global transitivity, and degree assortativity) within the context of the domain to generate latent knowledge. Two examples of applied contextualisation within GCND are using supply chain inference to enable the identification of "Super-brokers" and the prediction of attitude.

The “Super-broker” metric is designed to detect those entities that potentially provide brokerage services connecting entities involved in adjacent phases of the supply chain. For example, connecting entities that have trafficked drugs to those entities that wholesale drugs. The value of the “Super-broker” metric, whilst undoubtedly useful from a theoretic perspective, can only be ascertained by experts deploying the metric in real-world applications and measuring the impact of utilising this metric in strategic and operational contexts. From a runtime performance perspective testing on the fused dataset (9m node graph) sees the “Super-broker” metric run in ~ 27 seconds.

The RWAP attitude prediction metric is designed to utilise an incomplete set of observed attitude scores (for example, derived from conviction data) and use a propagation algorithm to impute attitude scores across the network to discover unobserved scores of nodes. Testing has shown that utilising local information (observed attitude scores) in conjunction with global information (community mean attitude scores) creates the possibility of quality attitude predictions. Specifically, using leave one out analysis to assess performance in relation to reachability we see that RWAP achieves correlation of 0.59 – 0.78 of masked observed scores when they have a direct neighbour (N_1) that has an observed attitude score, a correlation of 0.14 – 0.44 of masked observed scores when they have an indirect neighbour two hops away (N_2) that has an observed attitude score and a negligible correlation beyond two hops.

These results illustrate the difficulty of predicting imbalanced behavioural elements without local information. However, from an applied perspective testing indicates we can utilise information from a small fraction of entities and make quality predictions in 3 to 4 times as many directly and indirectly associated entities (N_1 and N_2). These associated entities are highly likely to be representative of a significant portion of criminal entities, dependent on data incompleteness and quality, as the likelihood of persons with criminal attitudes coalescing is high, if not logically critical for successful criminal profit generation.

This knowledge also gives clear indication of areas in the graph where prediction is poor or not possible and can be targeted topologically for data collection, and in the process measuring the performance of RWAP and improve its predictive performance. From a runtime performance perspective testing on the fused dataset (9m node graph) sees the RWAP metric run in ~ 58 seconds.

The “Super-broker” metric and RWAP have been developed into a wrapper function written in R and contained within the closed source R package KnowledgeDiscovery.

This Contextualisation module supports a complex systems view enabling the discovery of latent knowledge across various levels across the microscopic, mesoscopic and macroscopic spectrum. Let's now focus on the novel graph mining approach "GraphExtract" (Robinson & Scogings, 2018) that has been developed to detect criminal subgraphs and enable these metrics to be applied from a complex systems perspective.

5.3 Microscopic, mesoscopic and macroscopic knowledge discovery

Although previously we have separated microscopic, mesoscopic and macroscopic knowledge discovery into distinct areas, the reality in applied settings is that they are just differing perspectives of a system on a spectrum that runs from the entity level through to the entire network or system level. The "GraphExtract" algorithm generates multiple perspective knowledge discovery – from the entity through to subgraphs and the mesoscopic representation of subgraphs through to the entire network – and as such it makes more sense to convey the microscopic, mesoscopic and macroscopic spectrum as one connected concept.

The following section (5.3.1) is taken in part from the paper: Robinson, D., & Scogings, C. (2018). The detection of criminal groups in real-world fused data: using the graph-mining algorithm "GraphExtract". *Security Informatics, 7* (2), 1.

5.3.1 Detecting criminal groups using the "GraphExtract" graph-mining algorithm

Viewing the criminal domain as a system, and more specifically as a supply chain based system can provide the basis to computationally infer the potential path illicit proceeds take from the proceed generation phases (Manufacturing, Trafficking, Wholesaling and Retailing) through to the vehicle(s) that laundering those proceeds. At a micro level this provides the basis of prioritised leads (starting points for the intelligence or investigation process) for follow-up. At the meso and macro levels a deeper understanding of the potential opportunities available to launder money is made available. Comparison of the modus operandi available to differing functional and formal groups is critical to understand which groups are more sophisticated and use what methods predominantly. For example, some groups will look to shift proceeds through the purchase of assets such as motor vehicles and real property, whereas others may use corporate structures and the non-transparency provided by tax havens to conceal and then channel back proceeds via false invoices to associated business interests. Others will simply use the informal or formal banking system. Knowledge of how groups are

laundering gives law enforcement the ability to target the system and introduce measures to inhibit these mechanisms. For example, if company structures are being used the accountants and trust and service providers responsible for the formation of those companies can be targeted for attention, and changes to legislation to compel a higher level of due diligence could be required. If informal or formal banking systems are being used the Financial Transactions Reporting Act (1996) can be used as a basis to provide more education on "financial institutions" obligations and/or impose sanction for failure to adhere to obligations.

The graph-mining algorithm "GraphExtract" – as detailed in Robinson and Scogings (2018) – has been developed to identify generic fragments of profit-driven criminal activity enabling the extraction of relevant criminal subgraphs. As discussed, these subgraphs can provide a basis for generating investigation leads, and understanding how criminal groups operate and inter-relate as a complex system using different structures or typologies.

The range of computational methods applied by law enforcement and intelligence agencies on the detection of crime include supervised learning, unsupervised learning and anomaly detection (see 2.3.1 graph mining literature review). The deployment of these methods is typically conducted on focussed aspects of crime where there is a mature body of enterprise experience and a large scale of relative high-quality data. The problems are usually framed in a closed world approach where the problems are constrained to a restricted abstract level bound by what data is immediately available. Examples include the prediction of group's involvement in a specific financial crime or using entities to seed the problem for shortest path detection (Harper & Harris, 1975; Schroeder et al., 2007).

The fundamental alternative to a closed world approach, is the adoption of the open world stance. The open world assumption is the premise from which the "GraphExtract" algorithm has been developed. "GraphExtract" assumes that criminal activity is conducted within a complex system, and attempts to detect fragments of crime within this complex system. The multiple sets of data that give us a view of this complex system is characterised by being partial and contaminated. This contaminated partial view is derived from participants in criminal activity attempting to conceal their involvement and as an artefact of the error generated via the manual data collection and curation process. Misinformation comes in the form of fake entities, identity fraud, multiple aliases, use of fake addresses, multiple addresses, name variants, and using other proxy entities as a buffer against any digital audit trail (Maeno, 2009).

Framing criminal activity in terms of a complex system is based on the evidence that humans, whether criminally focussed or not, naturally coalesce into groups and interact within a broader complex system. The individuals within these groups, whether functional or not, influence one another, and present differing elements of capital and knowledge (Turner, 1991; Sparrow, 1991; McGloin &

Nguyen, 2013). Criminal opportunities are identified and functional goals formed to exploit these observed opportunities for individual and group advantage. The aggregate group resources are then used to engage in collaborative criminal acts with each individual fulfilling specific roles (Klerks, 2001; Coles, 2001; Morselli, 2005; Malm, Bichler & Nash, 2011), all within the broader context of the entire complex system (Robinson & Scogings, 2017).

As there is a real lack of accurate data representing these concealed behaviours at an individual level we can utilise data across the entire complex system to aid the detection of atomic criminal events. The "GraphExtract" algorithm is designed to take the broader complex system into account to detect overlapping subgraphs of entities that represent fragments of atomic criminal events and additionally create a secondary representation of how these subgraphs are connected to give a contextual representation of the criminal dimension within the complex system.

The lack of quality and quantity data available coupled to the obfuscated nature of the real-world problem has implications in terms of expectations of what we can computationally detect, and the coupled uncertainty. Therefore, it is appropriate to explicitly state that the detection of atomic criminal subgraphs is on prima facie basis. Within the context of the intelligence and investigation life cycle, and indeed the broader judicial system, the detection of criminal activity is a very early stage. As such, criminal subgraph detection in this context will always require the iterative collection of further information and corroboration building sufficient evidence to reach a decision that the subgraph represents an actual criminal event or not, and to what extent the fragment detected represents the near complete subgraph. The phrase atomic is used as it is important to convey the sense that criminal events are interdependent and overlapping both in terms of being discrete criminal acts over time and as part of larger conspiratorial complex ongoing criminal activities, like drug manufacture, within a system such as the illicit drug supply chain (Robinson & Scogings, 2017). It is therefore important to understand and convey this interdependency by using the work atomic to make this explicit.

At this point the benefit of using a computational method to detect atomic fragments of crime is evident. Firstly, "GraphExtract" has wide applicability when compared against supervised learning approaches as it utilises less specific data, and does not require training data. Secondly, "GraphExtract" is relatively bias-free in terms of detection coverage covering the entire criminal spectrum rather than being exposed to the bias introduced through reactive pathways. An example of reactive bias is the relative success of CHIS approaches on some ethnic groups over others. Thirdly, the coverage across the entire complex system and the creation of another group abstract level enables the opportunity to make better contextual decisions across the intelligence and investigation life cycle. These decisions include what criminal events to focus on and how resource is applied at a micro (e.g.

surveil a specific entity lead), meso (e.g. deploy intelligence and investigation resource on a specific group), and macro (e.g. investment on a specific intervention resource) level. Lastly, the generic benefits of any well designed and implemented computational asset, when compared with a manual or reactive approach, also exist here. Benefits include repeatability, consistency, measurability, extensibility, scalability, efficiency and transparency.

So, to summarise, we have the generic goal of proactively identifying atomic fragments of criminal activity as early as possible in the intelligence and investigative processes from a collection of heterogeneous datasets of criminal and non-criminal entities marked by high incompleteness, misinformation and uncertainty that has been fused via entity resolution and link prediction. The explicit open world stance of assuming the observable data only represents a fraction of the real-world problem is at the generic wide applicability end of the spectrum. Especially when compared to closed world data-coupled specific supervised or unsupervised learning approaches designed to predict very specific instances of crime.

Having earlier outlined the goal and purpose of "GraphExtract", the related approaches, the law enforcement and intelligence agencies context, the complexities presented by the data, and the value able to be derived, the novelty and significance of the algorithm is clear. We can now move on to outline the "GraphExtract" algorithm, cover the baseline assumptions, go over the detailed design of the algorithm, and evaluate the performance of "GraphExtract".

5.3.1.1 "GraphExtract" outline

Given the context surrounding the utility of the "GraphExtract" algorithm we can now outline its basic design. The input to the algorithm is a fused property graph containing multiple vertex types (e.g. persons, organisations, phones) that is derived from a range of datasets covering both criminal and non-criminal entities. Let's refer to the input as the original fused graph. The vertices of the original fused graph are then labelled based on their role in profit-driven criminal activity. Labels represent the primary and secondary interdependent roles. The primary roles include "Predicate offence", "Associated offence", "Alleged money laundering offence", and "Potential money laundering vehicles". The secondary role is "Realisation of assets". Vertices are labelled as per their involvement in each specific role. For example, an entity involved in drug trafficking will be labelled "Predicate offence".

The labels of edges are then determined by whether they meet the criteria of being a non-trust or trust-based relationship (see below for detail).

The primary labelled vertices form the set of vertices known as the "entities of interest". The pairwise distance of the "entities of interest" is then generated and used to construct a weighted graph from which community detection generates non-overlapping partitions (see below for detail).

Mediating vertices for each partition of "entities of interest" are then identified and added. Each partition of vertices, comprised of a subset of "entities of interest" and their relevant mediating vertices, then forms the basis from which to extract induced subgraphs from the original fused graph.

The algorithm uses an iterating parallel radial walk that subsumes neighbours terminating as the subgraph reaches 150 vertices or at the point of reaching a graph distance of four from the originating seeds.

The subgraphs are then also transformed into a mesoscopic graph where each subgraph is represented as a vertex and the overlap of vertices between two subgraphs forms the basis of how weighted edges are attributed.

The "GraphExtract" algorithm ouputs a set of atomic criminal subgraphs (microscopic view) and a graph illustrating the connections between each subgraph (mesoscopic graph) creating a view of how criminal groups interact across the entire complex system.

Both the microscopic subgraphs and mesoscopic graph can then be subjected to further knowledge discovery generating a richer multi-dimensional contextual view of the criminal complex system. These data representations can then be visualised and presented to a range of users to assist in better decision-making (e.g. intelligence analysts, investigators, managers).

We can now build on this outline of the "GraphExtract" algorithm by understanding the set of applied assumptions that underpin its deployment.

5.3.1.2 Assumptions

No matter how generic the design of an algorithm there are always going to be baseline assumptions that need to be met when being applied to the real-world. It is important to explicitly recognise the context in which law enforcement and intelligence agencies operate. These assumptions emanate from the acknowledgement that these agencies operate from an open world perspective, where data is incomplete and uncertain, and scalability is required. We will now go over the key applied assumptions that need to be met.

The data collected must fundamentally represent the core elements from which the problem is comprised. There must be a mature understanding of the problem and the differing elements that

make up the problem, coupled to access to a range of datasets that create a useful representation of those elements. For example, if the problem is profit-driven crime a range of interdependent concepts such as assets, corporate ownership, financial transactions, relationships, income, and criminality become important elements to represent.

As we can see above, relationships and attributes are key concepts. An explicit data model to represent relationships and attributes is a property graph. Relationships, otherwise known as edges, are represented as a pairwise interaction (e.g. a person has made a financial transaction to a company). Attributes are a property of either an edge or an entity (e.g. the date of purchasing a motor vehicle; the gender of a person entity). So, we end up with a variety of datasets that fundamentally represent a range of relevant entity types like corporate entities, persons, phones, addresses, and bank accounts, plus the known relationships between these entities, and relevant attributes of both the entities and these relationships.

In terms of the level of richness or detail required, there needs to be some way to establish the involvement of a minimum set of entities in the illicit generation, laundering, and realisation of proceeds. A complete set of observations is not required, however, the higher the quality of the input data will inevitably lead to higher quality output. Often specific datasets will explicitly detail a specific concept. For example, a register of motor vehicle ownership and real estate ownership can be used to identify assets linked to specific entities at a point in time. These assets can be potentially derived through the realisation of proceeds. Corporate registers can be used to identify potential vehicles to launder money. Criminal convictions can be used to identify individual's access or means to generate illicit proceeds. Then certain datasets can provide a basis to map across the fuller range of concepts. For example, Suspicious Transaction Reports can provide insight in terms of assets, criminality, money laundering, and money laundering vehicles. Utilising these datasets together creates the opportunity to identify subgraph fragments that contain criminal entities (entities with access and or means to generate illicit proceeds), corporate structures (means to launder proceeds), asset ownership (assets potentially realised through illicit means), and suspicious transactions (indicator of money laundering).

So, the goal is to represent the problem as fully and accurately as possible, understanding that any data representation will be a partial "dirty" view of the real-world. This data representation will be used as the basis to identify fragments of atomic criminal events at the earliest opportunity to enable an informed decision on resource deployment.

The core technology that enables multiple disparate datasets to be fused is entity resolution. The adoption of the open world perspective assumes that even in the presence of unique identifiers across datasets there will be a number of real-world entities that do not have unique identifiers or are

duplicated within the data. In any case the decision to include or exclude a specific dataset should not be based on whether that dataset has the unique identifier or not. "GraphExtract" assumes that entity resolution predictions are represented as predicted edges with an edge attribute of uncertainty [0-1]. Representing entity resolution predictions as an edge makes the prediction explicitly visible from a modelling perspective and to the consumer of the visualisation. This visibility ensures that algorithmic decisions are corroborated by humans, ensuring false positives are minimised.

Within this domain the determination of what edges infer a relationship of trust is critical. This is because failure of identifying trust and non-trust relationships will lead to subgraphs that include entities that have no material relationship to the core criminal event detected, reducing the quality of the subgraph. To guard against this the graph data model and quality of data needs to allow the identification of trust and non-trust relationships. Non-trust edges are typically transactional, asymmetric, non-enduring and consist of no transfer of social capital. For example, an entity making a wire transfer via a money remitter does not infer a material relationship between that person and the money remitter.

Lastly, it is important to acknowledge that the consumers of output from this algorithm will be operating in a variety of domain contexts with varying goals. As such it is critical to keep the algorithm as generic as possible enabling the consumer to "close the world" utilising their subject matter expertise and context to make informed decisions based on what data has been presented to them, making it explicit where there is uncertainty (e.g. entity resolution predictions) and corroboration is required.

For these reasons the "GraphExtract" algorithm focuses on detecting subgraphs containing the minimal set of entities required to operate together to represent an instance of generating, laundering and realising criminal proceeds.

Let's now examine the design of the "GraphExtract" algorithm.

5.3.1.3 Design

"GraphExtract": Identify entities of interest (step one)

The first step is based on detecting vertices within the original fused graph that are deemed entities of interest. The mechanism to do this is by identifying entity's involvement in the abstract conceptual elements that comprise a profit-driven criminal event - inferred or alleged predicate offences, associated offences, money laundering offences, proceeds realisation, and the vehicles used to commit these acts (see Figure 5.7.).

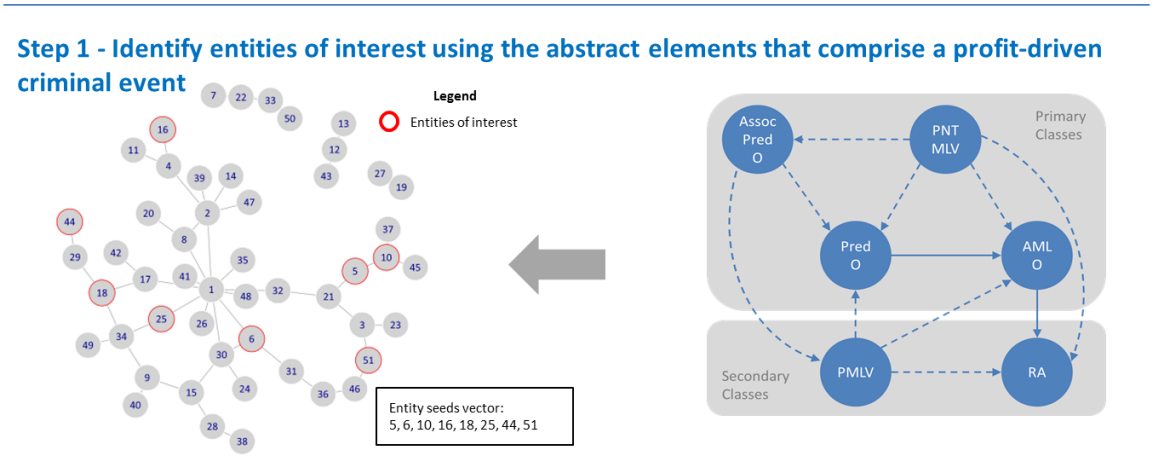


Figure 5.7. This figure illustrates the first step of "GraphExtract" – identifying entities of interest (this figure is a direct copy from Robinson and Scogings, 2018, p. 6).

'Predicate Offence' (i.e. "PredO") labels are applied in instances where an entity has an allegation or conviction in relation to a primary criminal act (e.g. drug trafficking). 'Associated to Predicate Offence' (i.e. "AssocPredO") labels are issued to entities that have a conviction or allegation that they engaged in a criminal act that was directly associated to the predicate offence (e.g. identity fraud). 'Alleged Money Laundering Offence' (i.e. "AMLO") is a class of node where there is evidence that an entity has direct involvement in a suspicious transaction. 'Potential Money Laundering Vehicle' (i.e. 'PMLV') is a class of node where entities are deemed to be vehicles, or associated to vehicles, that can be used to launder money. This class of node could be a domestic corporate entity or a shareholder of a domestic corporate entity. 'Potential Non-Transparent Money Laundering Vehicle' (i.e. 'PNTMLV') is a vehicle, or an entity associated to a vehicle, that is non-transparent and can be used to launder money (e.g. a corporate entity with bearer shares). 'Realised Asset' (i.e. 'RA') is a vertex that represents an asset, like a motor vehicle or real property, or an entity that is associated to that asset. We can then use these labelled nodes to identify the set of "entities of interest".

The "entities of interest" set of vertices is derived from selecting all entities that have a primary label, and all nodes that have a secondary label and are directly associated to a primary labelled node. Classes considered primary classes are 'Predicate Offence', 'Associated Predicate Offence', 'alleged Money Laundering Offence' and 'Potential Non-Transparent Money Laundering Vehicle'. Classes considered secondary classes are 'Potential Money Laundering Vehicle' and 'Realised Asset'.

Secondary classes of nodes (those labelled 'Potential Money Laundering Vehicle' and 'Realised Asset') are only deemed relevant when they are proximal to a node within the primary class. This is because they merely represent usual business elements and are only relevant if coupled to a primary class labelled node which is an inferred risk element.

Entity resolution is used to determine proximity between secondary and primary nodes. For example, when an entity identified as a gang member (originally sourced from criminal data) is predicted to be the same real-world entity as a shareholder of a company (originally sourced from the company register) then the shareholder is included as an "entity of interest".

This approach to creating the set of "entities of interest" maintains the flexibility and sensitivity to deal with missing data, whilst retaining the ability to extract atomic subgraphs of varying sub-types. Importantly, the dependency of the algorithm's successful performance on the base data sufficiently representing the core problem and the accuracy of entity resolution is quite evident.

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

Step two focuses on partitioning the "entities of interest" set of nodes and then adding relevant mediating nodes to each partition (see Figure 5.8.), leaving all other residual nodes classless (i.e. NA).

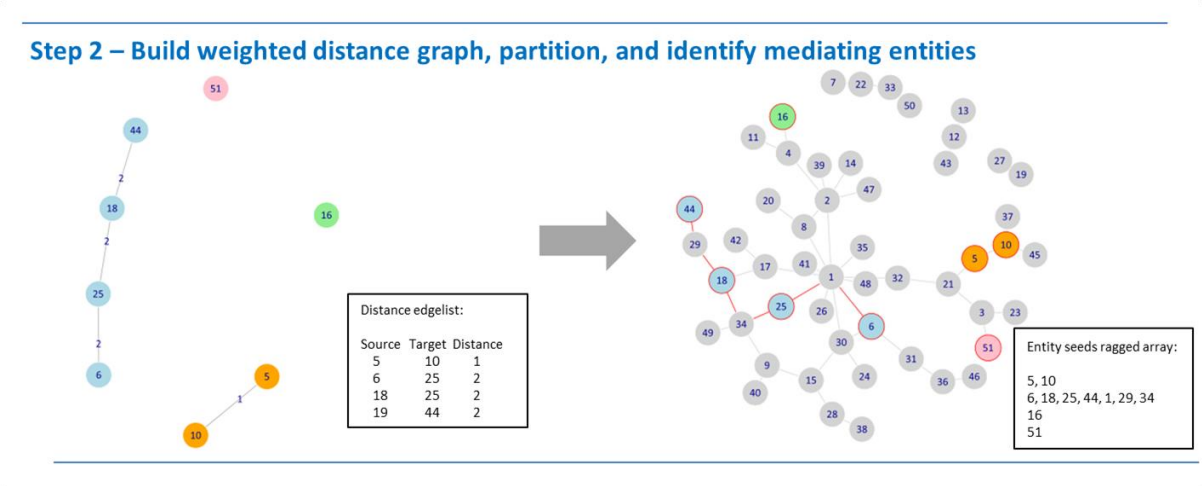


Figure 5.8. This figure portrays the second step of the process – identifying the seeds for subgraph extraction (this figure is a direct copy from Robinson and Scogings, 2018, p. 7).

The set of "entities of interest" collectively represent all those nodes across the entire complex system that are observed as being involved in profit-driven criminal activity. This second step utilises the set of "entities of interest" and attempts to divide the set up into groups that likely represent atomic functional groups of criminal actors/vehicles. A coarse yet elegant approach has been developed to mitigate the complexity and variety of criminal subgraphs, and the incompleteness and quality issues within the data. At this point it is useful to note that the complexity of criminal subgraphs is largely derived from the fact that criminal actors are dynamic and will form a range of criminally based relationships (Krebs, 2002; Morselli, 2010; Everton, 2013) over time generating overlapping functional groups.

The clustering approach is based on identifying subsets of "entities of interest" that have a small graph distance from one another in combination with a large distance to other subsets. To achieve this we firstly construct a weighted distance graph by measuring the pairwise distance (i.e. the length of the shortest path between a pair of nodes) between all "entities of interest". The maximum graph distance recorded is dependent on the data model used, however the default used here was 2. Adopting a maximum graph distance creates a scalable weighted graph, from which non-overlapping communities of "entities of interest" are extracted. This approach helps ensure that the extracted subgraphs are as atomic as possible, reducing the likelihood of extracting compound subgraphs that can result in nested structures.

The community detection algorithm used here was InfoMap (Rosvall & Bergstrom, 2008) based on the most appropriate granularity of communities in a reasonable time frame. However, experimentation on the best way to detect communities, whether non-overlapping or overlapping, in the most generalised sense is a priority extension.

The set of "entities of interest" is now divided up into a series of communities that represent a range of node label constellations. The constellations are comprised of various combinations of node classes that reflect differing criminal event fragments. For example, a community may include five entities. A gang member and a gang associate that both have previous convictions for the supply of drugs (i.e. "PredO"), a suspicious transaction of $10 million (i.e. "AMLO"), a stichting domiciled in Curacao (i.e. 'PNTMLV'), and an associated property (i.e. 'RA'). These five entities together form a specific constellation of class labels (i.e. "PredO", "AMLO", "PNTMLV", "RA").

As we know that the data is incomplete, and particularly our ability of correctly label nodes, we can then use the graph representation to identify nodes that are also potentially 'unobserved' entities of interest. This is achieved by taking each "entities of interest" community and identifying all nodes that exist in the shortest path between each node pair. This step is based on the premise that in the context of severe data incompleteness we can use the notion of brokerage and homophily to support identifying relevant nodes – using the shortest path metric to execute. Brokerage is based around the role of introducing and mediating relationships (Gould & Fernandez, 1989; Morselli & Roy, 2008) and homophily refers to human's predilection for forming relationships with others that are similar (McPherson et al., 2001). So, entities that lay in positions in between nodes that are "entities of interest" are likely to be relevant. These mediating nodes may be a disposable phone that is used to communicate between "entities of interest", a person that is also involved in the criminal event but is merely 'unobserved', or a company that has been used to convey business legitimacy.

We now have subsets of "entities of interest" and relevant mediating nodes that form the seeds to extract the subgraphs. These vectors of seed nodes are represented as a ragged array (see Figure 5.9.

step 3.). Conceptually each set of seed nodes represents a fragment of the atomic elements of a criminal event. The next step is designed to use these seeds to discover the relevant boundaries for each subgraph – boundaries that will naturally overlap on many cases (Morselli, 2010; Robinson & Scogings, 2017).

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

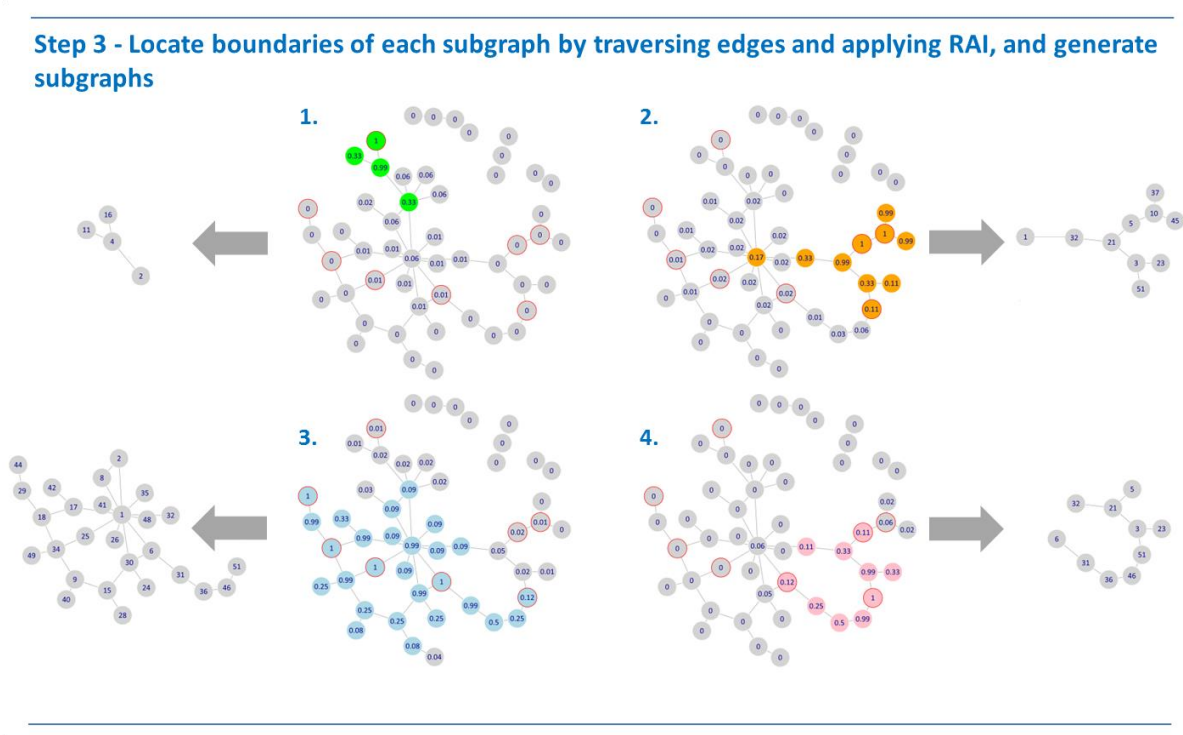


Figure 5.9. This figure illustrates the third step of the process to generate subgraphs – subgraph extraction (this figure is a direct copy from Robinson and Scogings, 2018, p. 8).

The third step involves using the seeds derived from each community (as per step 2) as a basis to yield induced subgraphs. Basic approaches to generate induced subgraphs generally use ego-based *n* neighbourhood extractions. "GraphExtract" utilises an iterating parallel radial walk that makes decisions on when a path should stop walking based on topological features encountered. This nuance enables a more precise and relevant representation of the nodes involved in the core atomic criminal event subgraph. Due to the impact the graphs topology has on the algorithmic decision-making alternate graph data models (e.g. bipartite graph; hyper edge graph) or significant differences in graph topology (e.g. not an approximately scale-free degree distributed) may require reconfiguring.

Each vector of seed nodes from the ragged array is used as the basis to conduct a parallel radial walk iteratively adding neighbours until either the subgraph has reached a size of 150 nodes or a distance of

four hops has been reached. If the walk reaches a supernode then that distinct walk terminates. A supernode is defined as any node that maintains a 'large' set of non-trust edges (e.g. money remitter).

Following the identification of the nodes that comprise the raw subgraph the Resource Allocation Index (RAI) (Zhou et al., 2009) is used as the mechanism to identify irrelevant nodes that require pruning. It measures "the product of the inverse normalised degree of all nodes along every shortest path between a pair" resulting in pairs involving longer paths and higher degree vertices scoring close to zero and pairs involving shorter paths and low degree vertices scoring close to one. In this way the high scoring more unique path-based relationships are prioritised over longer paths that involve 'popular' vertices, thus focusing on the more unique relevant paths and avoiding supernodes. RAI is conducted on all pairs involving the seed nodes, with each node's score based on their highest RAI score. Any node scoring under the pre-defined parameter (default = 0.07) are pruned from the subgraph (see Figure 5.9.).

Every functional group has a natural ceiling in terms of number of nodes. The subgraphs extracted, as loosely representing a functional criminal group, mirror this natural size limitation. The approximate upper ceiling of nodes within a subgraph is based on a number of elements. The key element is the evidence that social groups have an approximate ceiling of 150 (Dunbar, 1992; Hernando, Villuendas, Vesperinas, Abad & Plastino, 2009), which, along with the human-centred perspective that consumers of a subgraph visualisation can only reasonably deal with a small amount of novel detail when attempting analyse and interpret, and the computational performance of rendering complex subgraphs degrades rapidly as the size increases.

At this point it is important to remind the reader that the algorithm is specifically generic and attempting a more prescriptive approach would naturally limit its applicability across both domains and data of varying incompleteness. The goal is to identify a fragment of a functional criminal group to enable further risk management decision (such as information collection) given the uncertainty and competing priorities of other potential criminal events. The goal is not to detect the complete set of actors and their roles in a specific act, as we simply will not have the data to support such an assertion so early in the intelligence / investigations process.

Creating a series of subgraphs in this way generates microscopic leads which can be used to initiate the intelligence / investigations process.

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

To this point the focus has been to generate value that can be consumed at the microscopic level. Step four however focuses on creating a mesoscopic view of the subgraphs. This is done by creating a weighted graph – known as the mesoscopic graph – representing how each subgraph's set of nodes overlap with sets of nodes from adjacent subgraphs (see Figure 5.10.).

The mesoscopic graph is generated by representing subgraphs as nodes with edges representing when there is an intersection of nodes between two subgraphs. The weights of the edges are based on the ratio of number of intersecting nodes over the number of nodes in the smaller subgraph.

The mesoscopic perspective creates the opportunity to understand how each criminal subgraph relates to one another across the entire complex criminal network. This enables going beyond a descriptive aggregation-based analysis at the entity level to understanding potential emergent properties at the level of the group. Importantly the mesoscopic view provides the opportunity to generate knowledge discovery and understand the roles of atomic functional criminal groups in context and influence how prioritisation is conducted (Carley, 2006; Robinson & Scogings, 2017).

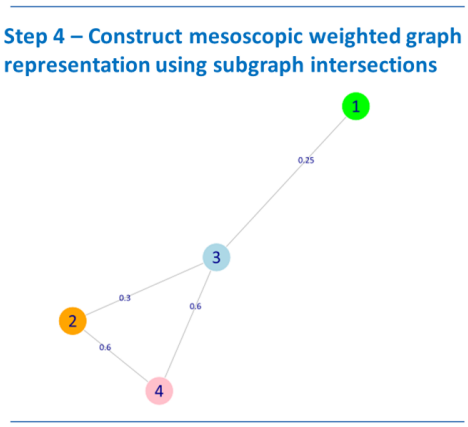


Figure 5.10. This figure illustrates the fourth step of the process - generate the mesoscopic graph (this figure is a direct copy from Robinson and Scogings, 2018, p. 9).

5.3.1.4 Utilising the output of "GraphExtract"

The output from "GraphExtract" consists of three data representations. This includes the original fused graph, the set of subgraphs extracted (microscopic view), and the mesoscopic graph that represents how each subgraph is interconnected. These representations form the basis to apply both knowledge discovery methods and expert domain knowledge to better understand the criminal

problem. A section of the mesoscopic graph with two subgraphs identified and exploded into the subgraph view is illustrated in figure 5.11.

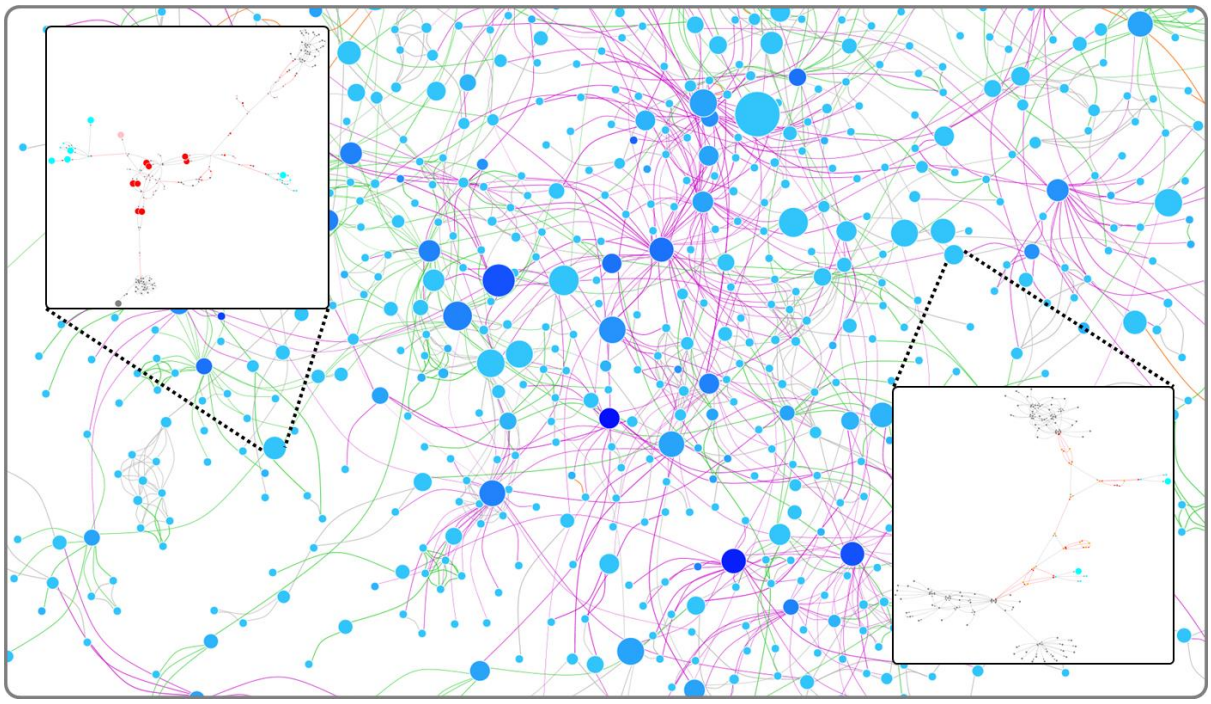


Figure 5.11. This figure illustrates a section of the mesoscopic view of the network with, two subgraphs exploded out into a microscopic view.

The original fused graph is a valuable reference data artefact as it is comprised of various datasets that each represents a key concept that underpins the problem targeted.

The subgraphs can be used as a collection of intelligence or investigations leads. These leads are used as the kernels for investigations and intelligence functions to undertake risk validation and risk mitigation. Figure 5.12 displays a pane of extracted atomic criminal subgraphs. These subgraphs demonstrate several criminal event subtypes, based on the topology and constellation of elements presented. The six subgraphs illustrated in figure 5.12 will be used to supplement the above explanation of the "GraphExtract" algorithm design.

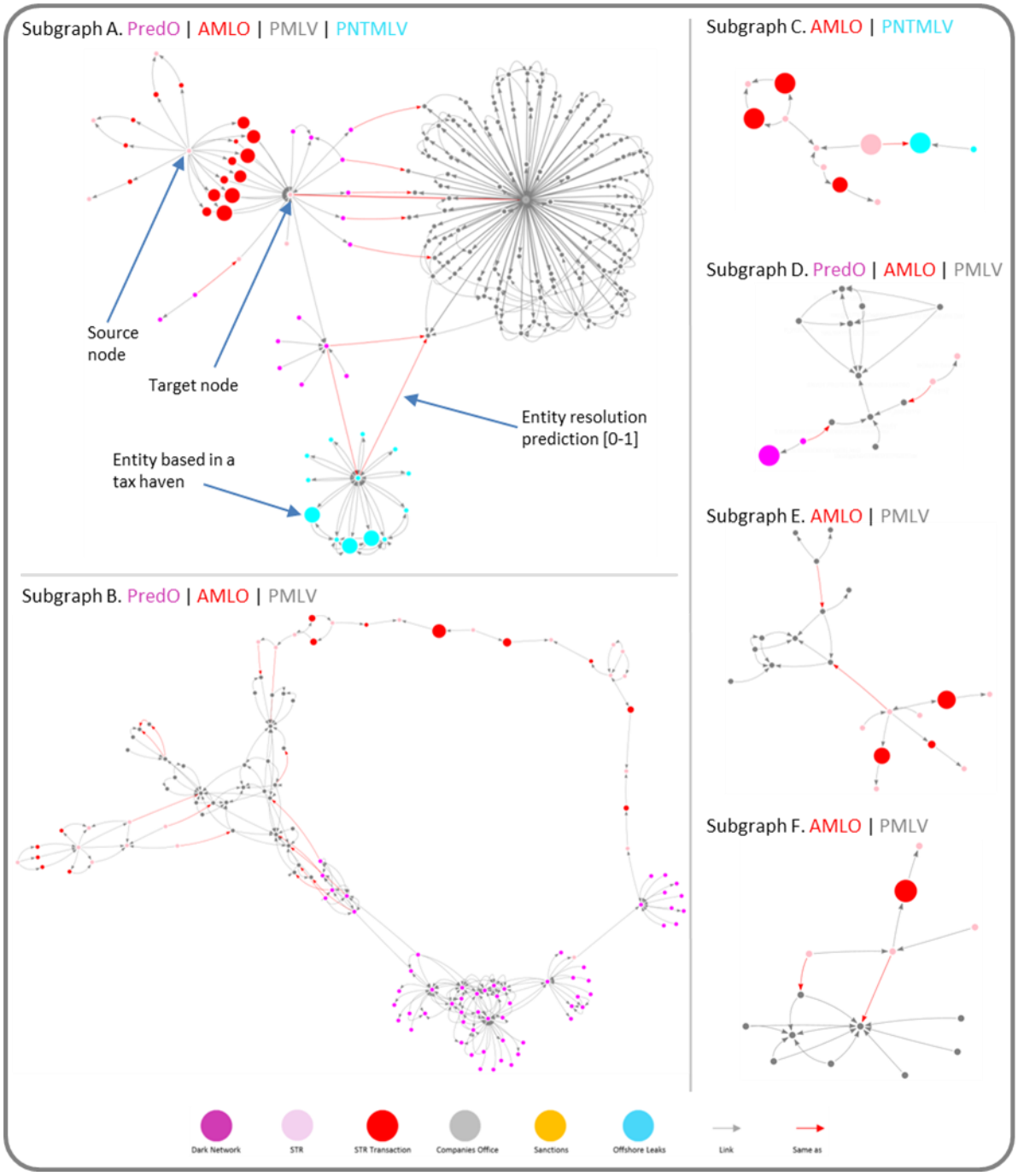


Figure 5.12. This figure displays a range of visualised subgraph examples, indicating multiple subgraph constellations (this figure is a direct copy from Robinson and Scogings, 2018, p. 12).

Subgraph A. presents as an approximately complete subgraph combining core elements (i.e. PredO | AMLO | PMLV | PNTMLV) of a criminal event. Subgraph A. depicts a collection of suspicious transactions (red nodes; AMLO) between the source node and the target node. The target node has a series of associations to organised crime entities (magenta nodes; PredO). The target node and six of the neighbouring organised crime entities have entity resolution predictions (red lines) to Companies Office entities (grey nodes; PMLV) indicating involvement in multiple corporate structures. One of

the adjacent organised crime entities also has an entity prediction with an entity from 'Offshore Leaks' data (cyan nodes; PNTMLV). The Offshore Leaks entities include three tax haven based corporate entities.

Subgraph A. portrays a real-world example of the domestic transfer of approximately $1 million to an organised crime associate that is a director of multiple domestic companies, and is connected to a series of tax haven based offshore shell companies.

Subgraph B. presents an approximately complete subgraph with a cyclic (C_{15}) topology, which includes three core elements (i.e. PredO | AMLO | PMLV) of a criminal event. This subgraph illustrates the non-trivial nature of consistently extracting relevant atomic criminal events from data that has severe incompleteness and complexity. At a simplified level we have a group of five organised crime entities (magenta nodes; PredO) on the left-hand side that are predicted to be the same entities involved in a company structure (grey nodes; PMLV) that have been the recipient of a series of suspicious transactions on the far left hand side. This organised crime controlled domestic corporate structure is connected to a broader cycle through a series of organised crime entities (magenta nodes; PredO) on the bottom and a chain of suspicious transactions (red nodes; AMLO) on the top.

The presence of the cycle and the unique pattern it represents creates the opportunity to collect more information to enhance the understanding of the wider activity of the subgraph in relation to the domestic corporate structure. The cycle may be an artefact of the data or the "GraphExtract" algorithm, failure in entity resolution or link prediction, represent a lack of data, and / or indicate an accurate real-world pattern of related criminal events that is yet to be explained. Maturing all these interdependent dimensions and understanding the conceptual and applied complexity of the problem are critical to evolving applied criminal intelligence approaches.

Subgraphs C and D provide examples of incomplete subgraphs. Subgraph C includes two core elements (i.e. AMLO | PNTMLV) of a criminal event but lacks the data linking this activity to the predicate offence. Subgraph D includes three core elements (i.e. PredO | AMLO | PMLV) of a criminal event but failed to fully traverse to include the suspicious transaction that is linked to the pink AMLO nodes.

Subgraphs E and F demonstrate the importance of uncertainty and in situ entity resolution predictions. Both subgraphs represent the same subtype with domestic corporate structures (PMLV) involved in sending domestic based suspicious transactions (AMLO). Entity resolution has predicted that an STR entity (the pink dot) represents the same entity as a shareholder(s) (the grey dot) of a domestic corporate entity. Each entity resolution prediction (red line) has an associated uncertainty attached;

however the context of in situ entity resolution prediction enables the corroboration of predictions. We see in Subgraph F the power of the logic of in situ ER with the two proximal entity resolution predictions buttressing one another reducing uncertainty.

These subgraphs can be presented in a number of ways with a range of supplementary metrics. The GCND solution includes the following vertex metrics with each subgraph; degree, brokerage role, supply chain role, "Super-broker", attitude prediction, affiliation, data provenance, data security, etc. Mesoscopic metrics are also provided including what constellation of criminal event elements is represented (e.g. AMLO | PMLV), number of non-transparent corporate entities present, associated countries, number of vertices, number of relationships, money laundering typology, transactional patterns, presence of graph cycles, offshore/domestic money laundering, offshore/domestic organised crime, and temporal metrics.

The subgraphs, and associated metrics, in combination with the mesoscopic graph can be utilised to understand the totality of how the criminal system operates and inform decision-making at mesoscopic and macroscopic levels. This solution creates the opportunity to generate a whole range of additional knowledge discovery metrics to enhance understanding. Law enforcement and intelligence agencies typically have strategic goals of 'dismantling' or 'disrupting' criminal or terrorist networks. The analogous complex systems concept is topological vulnerability. Topological vulnerability (or attack vulnerability) is basically the assessment of how vulnerable a graph or subgraph is to impaired performance under the threat of removing a set of nodes (Barabási & Albert, 1999; Holme et al., 2002). Topological vulnerability is related to the concepts of robustness, redundancy, and network resilience. Criminal networks are a special case as each functional criminal group is constantly balancing the maintenance of an efficient network carrying minimum topologically redundant entities with a robust network that is resilient to attack (Simmel & Wolff, 1950; Carley, 2003; Carley, 2006; Morselli, 2009; Robinson & Scogings, 2017). In a more concrete sense criminal groups are exposed to constant risk from both government agencies and competitors making trust a valued commodity. Additionally, these same criminal groups are dependent on getting requisite access to resource and knowledge to enable the successful execution of criminal goals. This range of factors will organically manifest, in combination with many other factors, to form a topology that's fit for purpose.

Therefore, measuring which nodes and subgraphs are most critical to preserving the criminal systems performance highlights the vulnerability of the criminal system. This creates clear targets for law enforcement and intelligence agencies to deploy resource and achieve maximal system disruption at both microscopic and mesoscopic levels. Using the range of metrics discussed to date, such as

brokerage, supply chain role, and attitude prediction, in combination with domain knowledge can create the basis for evidence-based and informed decision-making.

5.3.1.5 Performance

The performance of "GraphExtract" is in fact the basis of measuring the applied performance of the entire solution so this assessment will be detailed in the solution evaluation chapter next. However, some specific elements in relation to the performance of the "GraphExtract" algorithm are useful to cover here.

The accuracy of the subgraphs reflecting real-world fragments of atomic criminal groups/events was determined through a testing regime with one law enforcement agency. Two subject matter experts from the intelligence and investigations functions deemed 90% of the 100 sampled subgraphs as relevant and material. There is however a wait of at least a year or two before we can validate this tentative success through a more rigorous methodology because of the natural speed of the judicial system. The accuracy and relevance of the mesoscopic graph was more problematic to measure because subject matter experts had not been exposed to such visibility of the entire criminal system at both microscopic and mesoscopic levels. Notwithstanding the core-peripheral structure with subgraphs taking prominent brokering positions reflected the expectations of subject matter experts. So, while we have early tentative indications of accuracy and material value it is important to note that broader multi-domain real-world testing within a more rigorous experimental setup needs to be conducted before any comprehensive evaluation of material value for law enforcement and intelligence agencies can be made.

Measuring the computational performance of the "GraphExtract" algorithm will focus solely on the runtime and RAM expense. The next chapter will fully survey the wider performance metrics of not just the "GraphExtract" algorithm but the GCND in its entirety. The rationale is that the "GraphExtract" algorithm is the last stage of the GCND solution and is dependent on the performance of prior modules of Entity Prediction, Link Prediction, Contextualisation and Partitioning. So, it makes sense to measure performance in this context within the next chapter.

The test dataset is comprised of the four evaluation datasets entity resolved into one fused graph (entity resolution was conducted with a sampled F-measure 0.996). This resulted in a simplified graph of approximately 9 million nodes and 50 million edges represented as an igraph (Csardi & Nepusz, 2006) object. The set of subgraphs were outputted as a list of igraph objects. In this instance approximately 20,000 subgraphs were created in addition to the mesoscopic graph (also represented as an igraph object) in approximately 140 minutes runtime. The four steps each consisted of

approximate runtimes of 2, 20, 58 and 22 minutes respectively, with a mean runtime to generate a subgraph in step 3 of 0.1738 seconds.

"GraphExtract" was deployed on a Windows 10 environment with a CPU employing Intel Xeon @ 2.20GHz (8 cores) and 64 Gb RAM coded in the R language. Figure 5.13 depicts the computational expense of a random sample of 1,000 subgraphs. The axes relate to the number of nodes in each subgraph (x axis) and the number of edges in each subgraph (y axis), with the size of the dot representing the amount of RAM expended and colour representing runtime (in seconds). We can see the variation of subgraph size and how that relates to runtime and RAM expense. Runtime volatility is largely due to natural computational random fluctuations.

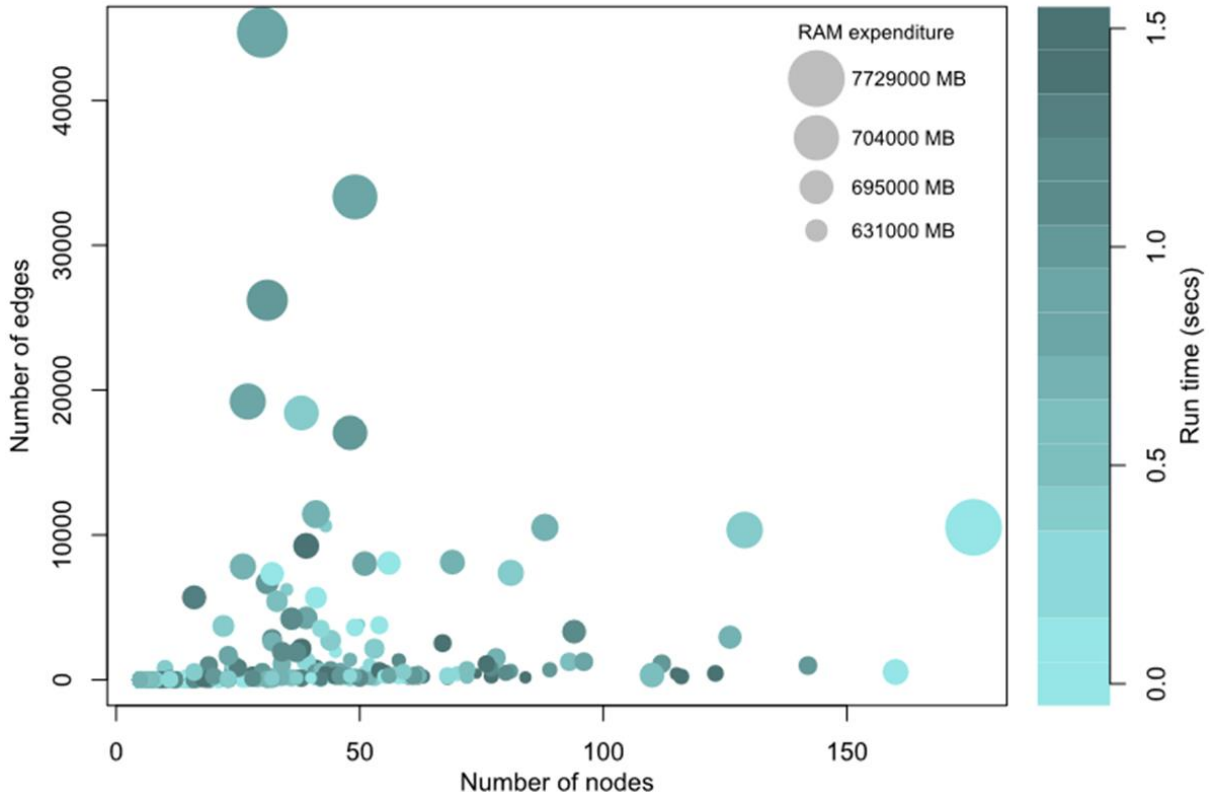


Figure 5.13. This figure depicts the computational expense of step 3 across a sample of 1,000 subgraphs (this figure is a direct copy from Robinson and Scogings, 2018, p. 11).

It is important to note that the computational performance metrics do not reflect optimised fully engineered production-ready software and is provided to give a tangible benchmark for speed and scalability of the implemented algorithm. Furthermore, it is key to note that "GraphExtract" is designed to be implemented in distributed computing context enabling future implementations to be scalable on big data.

5.3.1.6 Conclusion

The generic approach of the "GraphExtract" algorithm as detailed above adopts the complex systems paradigm. The outputs from "GraphExtract" have been demonstrated to show significant applied value across many perspectives simply by endeavouring to understand actors in the context of the system they operate within.

The first perspective is the detection of fragments of profit-driven criminal events. These fragments, represented as subgraphs, are fundamentally partial representations of functional groups of criminal entities that can be expressed as proactive prima facie 'leads'. These leads can be used as the seeds for intelligence or investigative resource substantiating the materiality and veracity of the risk, and subsequent risk mitigation. Beyond the ability to detect these functional groups there is also a wealth of latent knowledge in relation to the subgraph made available to enable enhanced understanding and a better ability to prioritise risk.

The second perspective is the ability to see how these functional groups inter-connect to form the entire criminal system. This view creates the opportunity to create latent knowledge on the system and on the groups within this unique context. The ability to generate knowledge as these levels creates the basis to make quality informed decision-making at meso and macro levels.

The third perspective is the explicit acknowledgement that generating a systems view of a problem and having developed expressive data representations of that problem from a complex systems perspective enables a systematic approach to evidence-based decision-making. Systematic because there is now evidence to support decision-making at micro meso and macro levels, creating the opportunity for coherent strategy flowing through to resource deployment at the more strategic level through to the meso and micro levels. For example, now that a systems view exists a series of core bundled concepts and the metrics to measure these concepts can be developed. The concepts could include vulnerability, performance, violence, use of corruption etc. all of which can be measured using a range of metrics with an associated degree of uncertainty. Importantly these strategic or macro concepts can be used to generate deployable strategy, like "the inhibition of the criminal system through the targeting of groups and individuals that perform scarce key roles like acquiring or brokering access to knowledge and resources". These strategic statements can then be directly translated into both meso and micro decision-making. At the meso level this strategy could be translated into improving the agencies data, ability to measure the core constructs, apportioning a larger resource on topologically vulnerable areas of the system at the group and entity level; focusing on trafficking and the diversion of pre-cursors; and on the core mechanisms used to launder money (e.g. real property; company structures). At a micro level this may influence investigations to focus

more effort at the facilitators of criminal activity rather than the most observable entities. Therefore, decision-making can evolve from a piece-meal disjointed approach to a more coherent aligned collective decision-making model.

In terms of performance "GraphExtract" has been subjectively assessed by two subject matter experts. These experts concluded that 90% of the sample of 100 subgraphs presented to them were relevant and the mesoscopic graph was consistent with their expert opinion of how criminal groups are connected. This must be couched in the context that the material they assessed is all previously unknowable and so a range of bias exists. The computational efficiency and scalability of the model indicates utility in graphs up to a size of ~ 9 million nodes and ~ 90 million edges with a runtime of around 140 minutes, performance that is far from prohibitive for many criminal domain applications.

"GraphExtract" has been developed into a wrapper function written in R and contained within the closed source R package KnowledgeDiscovery.

5.4 Summary of Discover Knowledge

Utilising contextual graph theory metrics, beyond the context-free application of standard graph metrics, hints at the value that a complex systems view can provide at the microscopic level. This value is then extended further by utilising the "GraphExtract" algorithm to accurately and efficiently identify criminal subgraphs from across the entire original fused data. These derived data assets provide the opportunity to generate a range of microscopic, mesoscopic and macroscopic views enabling domain experts to analyse a range of micro metrics in differing contexts and also create the opportunity to apply real systems thinking to new perspectives, including topology (network resilience) and assortativity.

The discovery of knowledge is a very broad and context dependent activity. The adoption of generic aspects in combination with contextual elements, plus a mix of network metrics and algorithms that are firmly rooted in a theoretical and empirical research provides remarkable insight across the micro meso macro spectrum. A core element of value is the high face validity that each element in isolation and in combination provides, making the totality of the output an elegant solution to inhibit criminal systems and particularly those "high hanging fruit".

Solution Evaluation [chapter 6]

Each module and sub-module, where applicable, has been analysed on performance with details provided within each chapter. We encourage the reader to understand the performance and limitations of each module and sub-module to gain an appreciation of the dependencies, complexities and the potential of this solution. Within this chapter we will combine all performance related data and provide an overall assessment of the solution. This will be informative in terms of the strengths and limitations of the solution, giving the context for the next chapter – potential extensions – and for the real-world application of the solution.

The GCND solution is written in the R language and structured in a modular way contained within three R packages – EntityResolution, LinkDiscovery, and KnowledgeDiscovery. These R packages have been designed to generate and persist a variety of data assets in conjunction with retaining only the core data assets in memory. In this way the deployment of GCND as an integrated solution is highly configurable and adaptable to a range of architectures, maximising the opportunity to reuse any data assets generated. These three R packages output the following data assets:

EntityResolution package:

- fused property graph
- table of ER predictions with metadata

LinkDiscovery package:

- fused property graph with inferred and predicted edges
- table of inferred links and LP predictions with metadata

KnowledgeDiscovery package:

- fused property graph with inferred and predicted edges and attributes (e.g. “Super-broker” and Attitude metrics);
- list of subgraphs (property graphs) with attributes;
- mesoscopic graph (property graph) with attributes.

All the testing done here was conducted using RStudio 1.0.143 and R 3.4.2 on a Windows 10 environment with a CPU employing Intel Xeon @ 2.20GHz (8 cores) and 64 Gb RAM.

6.1 Evaluation of the Make Data Exploitable section

The focus here is the contextual assessment of how exploitable the core data asset (original fused graph) is given the computational expense (runtime and RAM expenditure), and how that may compare to other products on the market (where possible).

The evaluation datasets include Sanctions, Dark Network/STR, Offshore Leaks and NZ Companies Office data. These four datasets were initially transformed into a harmonised data model – a property graph with a schemaless vertex attributes – with a series of cleansing functions executed to ensure each dataset was refined and compatible for fusion. Then each graph was independently entity resolved with the predictions used to contract each graph in turn. Table 3.1 details the before and after metrics of each of the four evaluation datasets, with table 4.9 providing details on the ER performance in comparison with a market competitor (where available). From this analysis we can clearly see that the EntityResolution package consistently creates better quality ER predictions across the evaluation datasets both in terms of the comprehensiveness of detecting 'same' entities (Recall) and the veracity of those predictions (Precision). The mean Recall across the three datasets where we had a comparison is 0.8520 and 0.9867 for the market competitor and the EntityResolution package respectively. The mean Precision across the three datasets where we had a comparison is 0.9259 and 0.9860 for the market competitor and the EntityResolution package respectively. The average F-measure across the three datasets where we had a comparison is 0.8723 and 0.9863 for the market competitor and the EntityResolution package respectively. It helps to translate these metrics into real tangible differences by examining the number of entities contracted. In terms of the Dark Network/STR and Offshore Leaks this translates into the market competitor generating a combined ~22,000 entities contracted in error versus ~2,000 by the EntityResolution package in conjunction with the EntityResolution package identifying an additional ~4,000 accurate predictions.

These four resolved evaluation datasets were then fused by amalgamating them into a disjoint graph and generating a set of ER predictions. The set of ER predictions were then used to fuse the disjoint graph representing the predictions as coloured edges (red) and creating the 'original fused graph' with an F-measure of 0.996.

At this point we have the four evaluation datasets that represent differing conceptual elements (assets, criminal risk, corporate structures, and non-transparency) of the problem domain (profit-driven criminal activity) fused into a single harmonised data asset that is entity resolved to a high level and represented in an explicit format (property graph) – the 'original fused graph'.

The consistency of the EntityResolution packages performance across the evaluation datasets, in contrast to the inconsistency illustrated by the market competitors, points towards a highly generalisable computational solution. Performing consistently well across a variety of datasets marked by varying completeness, quality, error, size and topology.

Now we turn to the issue of completeness, and particularly edge completeness. We take the 'original fused graph' and use the LinkDiscovery package to generate a set of inferred and predicted edges. This set of inferred and predicted edges represent instances where the LinkDiscovery package uses inference and machine learning to predict a real-world relationship exists where there is no observed edge in the data. LI is based on using logic and the explicit graph representation to infer relationships. This has been deployed in the form of a transitive based approach where any instances where a set of up to three persons are associated to an address or a company at the same time – and that address or company is not deemed a supernode – then it is inferred that a relationship exists between that set. Applying LI to the fused graph led to ~ 742,000 inferred relationships, with a very minimal set of errors (perhaps derived from data error, misinformation, the use of nominal actors, etc). Applying LP to the fused graph leads to a further ~3,000 predictions, with ~1,200 of those weak ties. Weak ties are those difficult to predict relationships where there is more than one mediating node laying on the shortest path between the pair. These weak ties are considered high value as they are more likely to represent pathways to scarce resource and these predictions go beyond what is immediately obvious to a human when observing visualisations of the data. Of this set of link predictions accuracy has been estimated to be in the range of 0.78 to 0.84. This is based on the sample-based testing done across the four evaluation datasets and the fused representation. The inferred and predicted edges are represented as coloured edges (blue) in the 'original fused graph'. Comparing the performance of the link prediction component of the LinkDiscovery package against other comparable models, we achieve good accuracy, speed and scalability. However, it is important to note that direct measurement was not conducted and that comparison is relying on published performance metrics on link prediction in the broader domain by Rhodes and Jones (2009), Fire, and co-authors, (2013), Lu, and co-authors, (2015), and Berlusconi, and co-authors, (2016). Unfortunately, scalability, computational speed, and the generalisability of these comparative link prediction approaches was not published.

So now we have a fused graph that has edge incompleteness improved via an additional set (~745,000) of highly accurate inferred and predicted edges.

At this point it is useful to explicitly outline the runtime to this point. The EntityResolution package takes approximately 17 hours to run on the four evaluation sets serially, and then an additional 12 hours to generate the fused graph. The LinkDiscovery package takes approximately 2 hours to run on

the fused graph. So, the "Make Data Exploitable" section takes approximately 31 hours to run, taking the four evaluation datasets and conducting entity resolution and link prediction resulting in a fused graph. The four evaluation datasets originally comprised around 18 million nodes and 93 million edges which subsequent to ER and LP results in a fused graph of approximately 9 million nodes and 90 million edges.

The scalability of the entire section has only been tested with the explicit data mentioned and the computational resource outlined above. Scalability beyond these sizes of datasets remain untested, however as mentioned earlier most core functions created have been designed to enable deployment within a distributed architecture at a future point in time.

Other elements of evaluation include the generic applicability, transparency, extensibility, and usability of the modules. The generic applicability of the modules was a high priority feature. The goal here was to create modules that can be applied to any criminal dataset, given the scalability limitations. Even if the data has absolutely no relationships the module will run effectively and produce quality results. The modules have been tested across four disparate datasets, however further testing is clearly important. Particularly with ER the performance is dependent on the predominant proper name origin represented and how that set is represented in the secondary sources and how accurately the Proper Name Origin Classifier and the ER model in general deals with the unique features of that set. The transparency of results is explicit due to the exposure of as much metadata as possible. A key goal was to expose as much of the metadata generated as possible. This was because the metadata can serve as a specific data asset that can be utilised in various context-dependent ways. For example, the reference graph generated by the EntityResolution package may be useful in text engineering streams. Extensibility is a design feature via modular construction. Effectively each chunk of code can be extended and swapped out for a more performant version. At a less invasive level the construction of a series of wrapper functions at various levels enable users of various technical ability to utilise elements of the EntityResolution package in isolation and use these chunks to construct related computational solutions (e.g. real-time federated search).

6.2 Evaluation of the Discover Knowledge section

The Discover Knowledge section executes in a time of ~ 141 minutes ("Super-brokerage" in ~ 27 seconds; RWAP in ~ 58 seconds; "GraphExtract" in ~ 140 minutes).

Evaluating the performance of knowledge discovery is always difficult 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. Evaluation is further

complicated when contextual metrics are developed because even if performance is validated in terms of the metric measuring the intended concept, the concept itself is susceptible to being determined as low value or immaterial or simply not understood due to the user's subject matter expertise maturity. The contextual value of "Super-brokerage" and attitude prediction (RWAP) is clear from a logically argued position. "Super-brokerage" represents those persons that hold critical brokerage positions in terms of the illicit drug supply chain and attitude represents a core behavioural element that drives criminal decision-making. Subjective anecdotal feedback from subject matter experts convey tentative support for the accuracy and ability to discern the materiality of "Super-brokerage" positions, however this positive feedback must be taken in the context of potential bias and lack of expertise. The argument for the success of RWAP is far clearer as the performance metrics, both automated (correlation of between 0.9 and 0.94) and human validation (N_1 correlation ranges between 0.59 and 0.78 and N_2 correlation ranges between 0.14 and 0.44), are impressive and the value of the construct is historically proven.

The most important component of the Discover Knowledge section is the "GraphExtract" algorithm, as its successful deployment can not only create thousands of meaningful contextual fragments of real-world criminality, it is also a mechanism to understand the criminal network as a system enabling insight at a microscopic, mesoscopic and macroscopic level. "GraphExtract" performance is dependent on the performance of the "Make Data Exploitable" section. However as demonstrated the performance of the EntityResolution and LinkDiscovery packages is highly accurate so the conditions in which "GraphExtract" has been deployed should be more than satisfactory for high performance. So, let's examine in detail the performance of the "GraphExtract" algorithm.

The conceptual value of using a proactive approach to detect generic profit-driven crime rather than placing a reliance on reactive methods is evident. But how well does the "GraphExtract" algorithm achieve this goal, how computational expensive and scalable is the algorithm and critically how relevant and material are the subgraphs and the mesoscopic graph.

The computational expense and scalability have been detailed above (chapter 5.3.1.5 Performance). In a concrete sense "GraphExtract" has been successfully executed on the 9 million nodes and 90 million edge original fused graph (the primary output from the "Make Data Exploitable" section) in 140 minutes, generating ~20,000 subgraphs. The relevance and materiality of the ~20,000 subgraphs requires subjective assessment.

The methodology used involves two subject matter experts (an intelligence analyst and investigator) validating a sample of 100 of the ~20,000 subgraphs in a screening exercise. The sample was generated by identifying a larger pool of relevant subgraphs that involved organised crime entities and material sums of suspicious transactions, and then drawing a random sample of 100 subgraphs. The

methodology mimics the standard law enforcement and intelligence agency tactical process. Each subject matter expert was instructed to independently score the set of 100 subgraphs on complexity, relevance, and completeness. Any disagreement between the experts was categorised as 'unsure'.

When measuring subgraph complexity the two subject matter experts found that 88% of the subgraphs were acceptably atomic, 9% were too complex and 3% unsure. The 9% of overly complex subgraphs were assessed further and found to contain a combination of incomplete data, data error, real-world complexity, and "GraphExtract" algorithm failure. The two subject matter experts found 90% of the subgraphs were determined to be relevant, with 5% irrelevant and 5% unsure. The irrelevant subgraphs identified were due to a combination of data incompleteness and failure of the "GraphExtract" algorithm. In terms of completeness the two subject matter experts found 86% of the subgraphs were complete, with 12% incomplete, and 2% unsure. The incomplete subgraphs were found to be caused by a lack of data and / or failure of the "GraphExtract" algorithm. A lack of resource meant that false negatives were not able to be examined.

The wider applicability of the "GraphExtract" algorithm was hinted at as firstly the two subject matter experts used in the experiment performed differing roles in the law enforcement agency (Intelligence Analyst and Investigator) and focused on two different criminal areas – organised crime and serious financial crime. Secondly, in their opinion 8 of the 100 subgraphs evaluated were determined to be of relevance to share with appropriate law enforcement and intelligence agencies.

It is important at this point to underline the tentative nature of these findings as this evaluation is limited to two subject matter experts from a single law enforcement agency assessing a sample of 100 subgraphs. Undeniably further testing in a variety of circumstances is required before the generic value of the subgraphs extracted from the "GraphExtract" algorithm can be fully tested.

An important secondary element to evaluate is the mesoscopic graph. Specifically, it is important to evaluate the topology of the mesoscopic graph and how well the mesoscopic graph reflects the real-world. Again, there is subjectivity involved in this evaluation; however this should not dissuade us from attempting to construct a methodology that partially measures the value in a logical and defensible way. This evaluation is vital as if no discernible structure or identifiable features were able to be identified that mirror subject matter expert's expectation of the real-world then there is no basis at a meso level to attribute success. Success being the ability of the "GraphExtract" algorithm to firstly detect criminal subgraphs that represent atomic functional criminal groups and secondarily transform the set of subgraphs into a meaningful representation of the entire criminal system at a mesoscopic level.

The methodology underpinning the mesoscopic evaluation is based around providing a visualisation of the giant component, with contextual metadata, and structural analysis using blockmodeling to subject matter experts to ascertain whether the results mirror what is expected. The difficulty in this applied scenario is that there of course exists bias and the fact that the analysis is testing a phenomenon that is largely untested. The focus of the subject matter experts is therefore on the plausibility of the structure and features of the mesoscopic graph, given current knowledge.

Figure 6.1 depicts the giant component of the mesoscopic graph. In terms of metadata, the size of the nodes represents the total sum of the suspicious transactions within each subgraph. The colour of the nodes represents whether that subgraph primarily contains transnational organised crime entities (orange), domestic organised crime entities (magenta), offshore suspicious transactions (yellow), domestic suspicious transactions (grey), or other (blue).

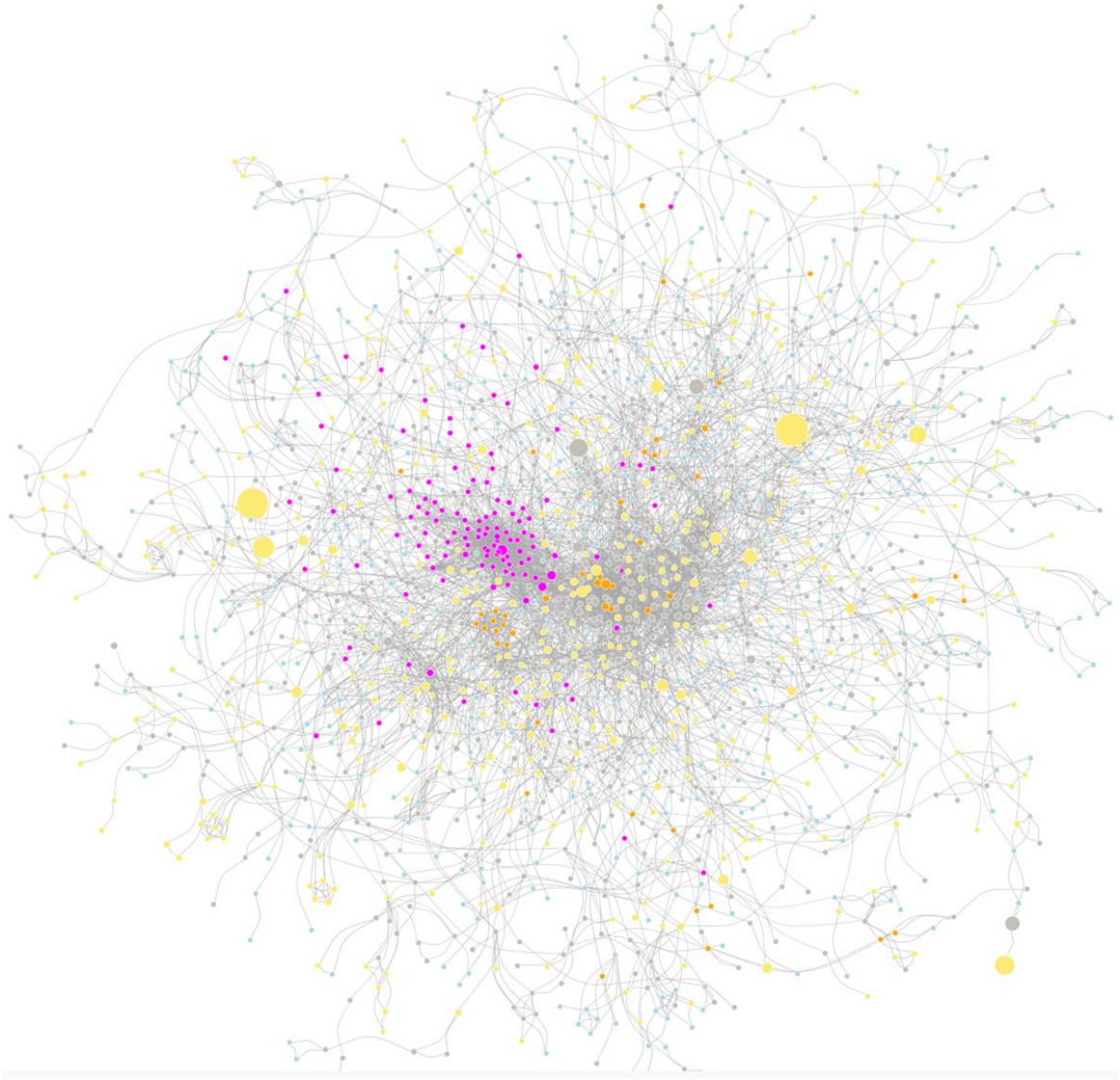


Figure 6.1. This figure gives an example of the giant component of the mesoscopic graph (this figure is a direct copy from Robinson and Scogings, 2018, p. 13).

The structure of the mesoscopic graph was analysed using blockmodeling. Blockmodeling is a well-established technique to assess structure (Wasserman & Faust, 1994) that has been applied on criminal networks (Xu & Chen, 2005).

Blockmodeling was applied to the giant component of the mesoscopic graph and found a core / peripheral structure with the core representing approximately 10% of the giant component's nodes. Figure 6.2 depicts the mesoscopic graphs giant components core divided into six distinct blocks and a peripheral seventh block that was not visualised. This was done via blockmodeling in an exploratory way selecting a variety of block classes until the optimal set of blocks was arrived at. Each block is represented by a different node border colour. The lower left block is represented by light grey node border, the middle left block (subgraph 20848) is black, the upper left block is yellow, the upper centre (subgraph 5164) is magenta, the upper right block is blue, and lower right block is lime green. The edge colour is based on the proportion of intersecting entities between subgraphs, with the subgraph with the lowest number of total entities in the dyad representing the denominator. Colouring the edges in this way aids the visual assessment of how related subgraphs are. Node size indicates the brokerage (Gould & Fernandez, 1989) of each nodes position, allowing easy identification of subgraph nodes 21302, 20848 and 5164 as the most prominent brokers.

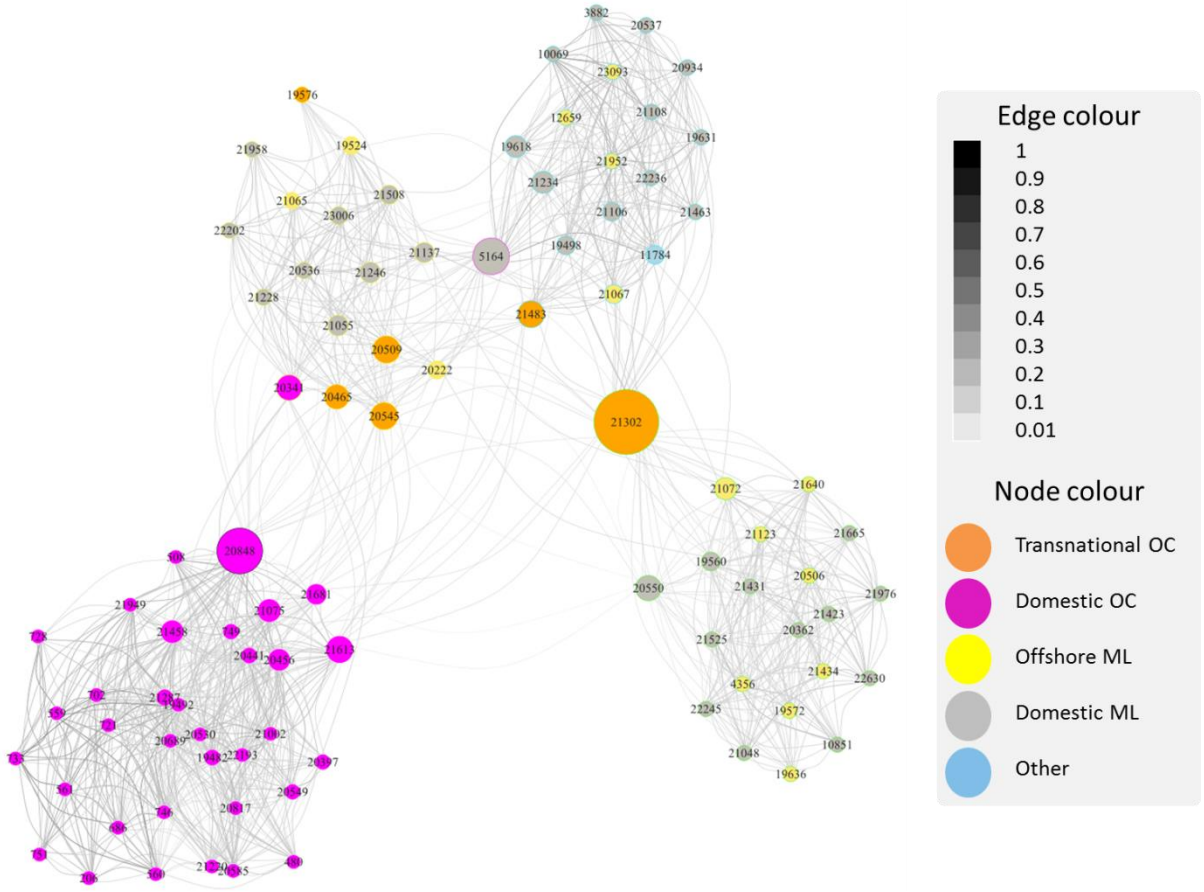


Figure 6.2. This figure depicts the six blocks (or classes) of the core of the mesoscopic graph (this figure is a direct copy from Robinson and Scogings, 2018, p. 14).

The findings from the visual analysis of the giant component of the mesoscopic graph in combination with the blockmodeling analysis indicate that the entire criminal system is represented by ~ 20,000 subgraphs. Within this criminal system there is an outer periphery of ~ 18,000 subgraphs signifying criminal activity materially disconnected to organised crime. Within the giant component there is a periphery of approximately 1,800 subgraphs that are indirectly associated to the core (~ 200 subgraphs) of the giant component that is dominated by organised crime. The core of the mesoscopic graph is itself brokered by a small number of subgraphs that have a significant organised crime involvement.

While the results broadly align with expectations from subject matter experts it is important to explicitly state that we must take caution with the significance of these results as the data is an incomplete representation of the real-world and as mentioned much more robust experimentation is required to validate and support the improvement of this approach. This does not devalue the work done to date but merely highlights the inherent limitations. However, having stated all of this there is clearly enough evidence, at both the microscopic and mesoscopic level, to indicate the current and potential significant value in the "GraphExtract" algorithm.

6.3 Summary

Evaluation of the “Make Data Exploitable” and “Discover Knowledge” sections demonstrates the discrete and collective value contained within the solution. A combined runtime of approximately 33 hours on a total of four disparate datasets combining for a total of 18 million nodes and 93 million edges generating a

- high quality fused graph with an entity resolution F-measure of 0.996 and an additional ~745,000 of highly accurate inferred and predicted edges,
- ~20,000 extracted subgraphs representing atomic profit-driven criminal events and associated contextual metadata, and
- a mesoscopic graph representing how each subgraph inter-relates to comprise the entire criminal complex system.

Subject matter experts have assessed both a small sample of subgraphs extracted and the mesoscopic graph and found utility and materiality, with 86-90% of subgraphs found to be appropriately atomic, relevant, and complete enough.

Part C: Potential Extensions and Summary

Part C outlines the immediate future for development possibilities and summarises the GCND computational solution.

Potential Extensions [chapter 7]

Extensions of the implemented modules will focus on three elements of entity resolution, link prediction, and knowledge discovery. Each of these implemented modules will be evaluated for potential extensions in turn.

7.1 In general

The current deployment of GCND is coded in the R language and contained as a set of R packages. Engineering the modules of code into a production state enterprise-ready software product would substantially improve usability, run-time and memory use. A range of approaches could be adopted to increase the model's use of parallelisation and distributed computing technology. Modules of code, where possible, have been developed with parallelisation and distributed computing in mind. For example, vertex contraction has been developed in an independent table based serial approach that can easily be configured to a distributed context. Likewise, some key graph metrics including graph distance and RAI have also been coded in table structures (e.g. triples/edge lists, vertex attributes, edge attributes) enabling a table-based approach like Apache Spark to be utilised.

7.2 Entity Resolution

The framework of the ER model is now well established and contains significant room for detailed extension. The results already are very impressive, so any further accuracy gains will be small and incremental. Furthermore, accuracy is always a trade-off with other aspects such as generalisability, complexity, speed and scalability. So now let's investigate some potential extensions.

Pre-processing is a good place to start. The relationship between ETL (extract transform load) and ER is complex. Many ETL elements, particularly cleansing and harmonisation, have been introduced into the ER model to ensure that the performance of the ER model is consistently high even when the input data has some inconsistencies (e.g. encoding, date formatting, date reconciliation, treatment of NA, case, treatment of spaces, treatment of special characters, harmonising prefixes and suffixes,

etc.). One such area is address parsing. The quality of addresses is very important to the success of ER on complex non-obvious cases. So, a simple step to improve ER performance is to parse addresses accurately prior to using the ER model. Address parsing is however an area that could be integrated quite easily into the model, much in the same way as the country identification algorithm has been. Enhancements to how addresses are represented maximises the ER model's ability to detect duplicate addresses. The treatment of units and apartments, rural addresses, PO boxes, overlapping suburbs, and the way countries are represented are a list of address related cleansing and harmonisation elements that can be improved and perhaps fundamentally shifted from a string to a geographical representation to enable more geographically based concepts being applied (e.g. propinquity). Having said all of this, the Entity Resolution module is not a comprehensive ETL model and the expectation is that ETL is performed prior.

A focus has been to ensure the model is not over-fit and performs well across a heterogeneous set of data. However, non-generalizability within ER is largely derived from a lack of access to significant entity datasets from different regions across the world, from which to derive secondary sources of data which in turn is used to generate the Name Origin Reference Graph. The current approach to improve generalizability is based on iterative development and attempting to derive abstract knowledge to generalise knowledge. A core example of this is the way the ER model uses the Name Origin Reference Graph, as an element of the PNOC, to identify homogeneous subsets of names from a common origin. Development of the ER model with this feature enables a more tailored identification of blocks, not only speeding up computation time, but also resulting in a more tailored pairwise assessment of equivalence. For example, the ability to identify Chinese origin names and treat them separately from Arabic origin names is of clear utility – as their name structures and features are quite different. At an abstract level the differences between the names of the major name origin partitions can be quantified linguistically (e.g. character length, word number, word length, syllables, etc.) and used to determine the optimal way to determine equivalence. So, for instance the pairwise assessment of Name Origin Reference Graph partitions that are similar (e.g. Subcontinent and Arabic origin name partitions) can be conducted together and automatically determined based on metadata. This approach has derived significant value thus far, and as the approach is generalised further we would expect additional improvements coming in incrementally diminishing returns as the most common and divergent name origin partitions are prioritised.

The Reference Graph Algorithm (RGA) has the potential to form the kernel of a knowledge graph where a range of relevant attributes (e.g. a range of similarity measures) are added to generate better performance. In instances where ongoing ER is taking place there is a clear benefit to persisting the Reference Graph and over time augmenting the graph with deltas. Manual curation is another potential area to explore with the aim of optimising the knowledge representation and as a by-product

convey potential improvements. Alternative approaches to create better scalability and improve runtime are key as the RGA currently consumes a significant portion of computational expense.

The Proper Name Classifier (PNC) is built on a simple premise, however there is plenty of room to develop this model further and apply to the ER model in a variety of ways. The first key decision within the model is how to define what a proper name is and apply this definition to the extraction of a subset of names in the data. The current definition is based on experimentally derived cut-off of all person names that have been used in 20 or more unique names. Development of a more advanced accurate method here needs to be explored. The second key decision is what edit distance algorithms to use and what thresholds or mechanism to define what names are different names. The third key decision is how to optimise the way in which we probabilistically identify real proper names that have been identified as being generated from typographic error (e.g. “Brian” versus “Brain”). A heuristic method is currently deployed based on a Bayesian approach, however there remains more nuanced opportunities (e.g. statistical) to generate enhanced performance.

The Proper Name Origin Classifier (PNOC) is designed to classify the origin of a proper name, to support multiple decision-making points including meta-blocking. Exploring and iteratively developing the model will bring clear benefits. Experimenting with new engineered features and utilising new approaches with current engineered features will be fruitful. For example, further experimentation with a variety of community detection or partitioning approaches when determining classes for the Name Origin Reference Graph will lead to development opportunities. Experimentation with other machine learning algorithms (e.g. random forest and deep learning) or evolving the model into using iterative or reinforcement learning is likely to yield more comprehensive understanding of the problem and likely better results. However, investment in these aspects need to be weighed up in terms of potential value and the impact on the overall goal of the ER model versus all other possible extensions.

The iterative curation of secondary data sources for better performance is not so much an extension as a core ongoing human-centred task to contribute to the effectiveness of the model. Lisbach and Meyer (2013) demonstrate the linguistic complexity of entity resolution and the importance of the semantics of proper names. This emphasises the benefits of linguistics analysis of proper names and building data representations of names and associated metadata to enable more accurate entity resolution that is tailored to the domain. The data representation that this knowledge is stored within is another key decision. Ontologies and standard ontological representations (e.g. semantic graphs) have much to contribute. This is a substantial undertaking that should be iteratively developed over time balanced against other potential improvements. Improvement in secondary data sources will increase performance across a range of the models previously discussed.

Supervised learning approaches (e.g. machine learning) to exploit the metadata extracted through the models deployment is likely a fruitful avenue for exploration. To date SVM and RPART machine learning approaches have been created as options within the code. SVM's performance is characterised by high accuracy and slow runtime, whereas RPART is characterised by lower accuracy but fast runtime. The cost is totally dependent on the sub-domain of the user, hence providing both options. Development in this area will no doubt provide benefit, particularly if coded to perform in a distributed computing context. A specific focus should be based on the semi-supervised element of how the 'labels' for the training set are generated. The problem is fundamentally unlabelled so we need to generate labels in some automated way. The current approach utilises a logic semantic based approach, however this labelling mechanism could be developed further.

Collective ER likewise has much to gain from experimentation in a variety of settings and understanding what concepts are critical to measure and incorporate into decision-making. Again, simple thresholds are used to make decisions which whilst reduces the complexity of the model, leaves much room for optimisation using probabilistic approaches.

Uncertainty is a critical concept, which is currently implemented using a semantic based approach. Experimentation would be valuable comparing the efficacy of alternative modelling techniques such as entropic based measures. It is true that uncertainty is implicitly measured with probabilistic machine learning and statistical methods. However, there is value to explicitly measuring uncertainty that is derived from data error and incompleteness and model error so that there is enhanced visibility of the source of the model's uncertainty. For example, measuring the amount of data, the quality of data, and the provenance of data will enable a better entity resolution prediction.

In situ ER is an area that has much applied potential and is under-developed. Utilising the context of how ER predictions are applied has much value and scope for development. Experimentation and user feedback are critical to generating contextual understanding of the value of subsets of applied ER predictions, and how this value can be understood and utilised to its fullest extent. Simulation is another avenue which would provide another valuable feedback mechanism to not only measure the accuracy of ER predictions, but also use it to investigate areas of the applied graph for more predictions.

Vertex contraction is applied here in a very simple way, with only core attributes retained for auditability. A more robust representation is a semantic graph approach which can retain all raw entity attributes and materialise only relevant attributes to users using a range of attributes (e.g. provenance) and latent knowledge (e.g. string length, element frequency, PNOC) to make better contextual decisions. Semantic technology should be explored further.

The Entity Resolution module is designed to take advantage of data in a graph format, however there is no barrier to other more traditional sources of data, such as tables, being used as an input. Also, the use of parameters to reflect the user domain-based requirements could be made available in a more interactive way (e.g. sliders) that then give a simulated set of predicted model results based on profiles of a sample of the data.

Increasing the visibility of performance can always be improved. The development of a range of automated metrics, beyond using global transitivity and diameter of non-transitive components found in the Prediction Graph, will further improve understanding of ER performance. Also, computational performance metrics could be improved to give a more granular understanding of how each module within the ER model performed. Developing an interactive graphical user interface such as been done in the D-Dupe software (Bilgic, Licamele, Getoor & Shneiderman, 2006) is a clear path to enable more accurate and faster human validation of ER results, and the collection of ideas for improvement. The current graphical output is HTML based, leveraging off the D3 Javascript library. However, much user experience related work is required to improve usability.

7.3 Link Prediction

The results derived here need to be repeated again on other heterogeneous criminal datasets of various sizes and topologies in different settings to give further empirical basis of how generalizable the link prediction models are. The fact that they are underpinned theoretically gives every chance for their performance to be generalizable however this remains to be fully tested. Additionally, in the real-world, different datasets will have differing issues of incompleteness and error, and it is yet to be determined how the models perform under a variety of different conditions.

Link prediction in its current abstraction comprises a set of heterogeneous sub-problems. These sub-problems include the prediction of missing links that are familial in nature, missing links that indicate an entities alias, and the identification of an entity fraudulently utilising another identity, amongst other sub-types. Other dimensions of the problem could include weak ties versus strong ties and the mechanisms underlying (e.g. homophily and cyclic closure) these sub-types. So, approaching link prediction for the criminal domain, at its appropriate abstraction so each subset is homogeneous is likely to generate significant value. To arrive at this level of understanding further exploration and development iterations are required. These iterations could include discrete sets of engineered features focussing on each lower abstraction of the criminal link prediction problem – such as familial edges, identity fraud edges, and alias edges.

The machine learning framework of the link prediction model is remarkably simple, enabling a raft of internal modification whilst not impacting on the framework. Feature engineering is an obvious area where existing features can be modified for speed, scalability and accuracy improvements, in addition to the engineering of new features. Other features to be experimented with include generalized blockmodeling (Doreian, Ferligoj, Batagelj, & Granovetter, 2005), overlapping community detection, and generalized topological overlap measure (Yip & Horvath, 2007). However, engineering may be required to ensure scalability.

In addition to employing a new set of engineered features we can develop new ways to generate explicit representation of the graph data that already exists. One approach is representing the mediating relationships in the path between the source and target node to generate additional context. For example, if we can accurately label existing relationships and then codify these existing relationships in terms of the path (e.g. familial – familial) would form a relevant new feature to assist prediction.

Any pairwise computational endeavour must deal with intractability. In this case we cannot computationally assess whether every pair has a non-observed edge and so we have to choose an approach to generate a subset of pairs on which to assess. The currently implemented approach is to select all pairs that fall within a shortest path of four "hops" (on a person unimodal graph). Dependent on the topology of the criminal network this approach may be too coarse (e.g. a network with substantial "small-world" features), so alternatives should be explored (e.g. community or name similarity) to enable scalability and preservation of accuracy across each sub-problem (e.g. aliases). From a domain perspective the users of the model may only be interested in a small set of entities (e.g. gang members and associates) who may only make up a very small percentage of the overall graph allowing that set to be compared with all other entities. Alternatively, the assortative community detection approach or a coarser community detection or component approach similar to network blocking strategies used with the entity resolution model could be used.

The current iterative nature of the model can be extended to include multiple models targeting specific sub-types of link prediction forming a more sophisticated approach that makes use of each models output. Reinforcement learning, or an analogous approach, can then be employed to utilise the multiple outputs and extend the value generated.

More specifically

Scalability on large graphs that exhibit an approximate scale-free degree distribution containing supernodes is a significant challenge, and so developing the techniques to solve this require further engineering.

Geographic similarity is likely to be a useful feature, and so development of a more nuanced measure to take this into account is likely to provide a generic benefit.

Enhancing the way temporality is modelled is critical to increase model accuracy whilst ensuring computational expense is minimised. But for temporality to be integrated to a high degree of accuracy there is a dependency on minimal data requirements and of course a flow on impact and opportunity across modelling decisions. These decisions need to be developed to enable a better temporal representation.

There is an opportunity to include an optimisation process so the SVM uses an optimised rather than hard-coded empirically derived cost parameter. Experimentation will illuminate the benefit and cost of including this specific optimisation routine.

Enhancing the link inference capability deployed will undoubtedly increase the performance of the LP model. A method to advance the currently deployed link inference approach is to develop an ontological approach that enables more fine-grained extensible inference making.

Generating a more explicit representation of the features generated through the model within the output, whilst managing computational expense. Outputting a more explicit set of relevant metrics within the output will enable better post-hoc analysis. This in turn will lead to a more enhanced understanding of link prediction and how the model works.

Alternate machine learning approaches could be substituted for the SVM model used, including Bayesian methods, regression, k-Nearest Neighbours algorithm (kNN), neural networks, or decision trees, to assess the strengths and weaknesses on an experimental basis. The constraints and requirements derived from the deployment of the model may influence the decision on which modelling approach is preferred, however the literature indicates SVM is empirically a consistent top performer.

Again, engineering the code into a more deployable language is important if the approach is to be deployed outside the lab in an enterprise context.

7.4 Discover Knowledge

Any number of directions can be taken with knowledge discovery as it is very domain dependent. The approaches noted above are deliberately generic and so the technology should be applicable across the criminal domain. Of course not all sub-domains will be interested in the drug supply chain, however the generic approach of mapping domain knowledge to the data is. Within the counter-

terrorist domain a number of tasks still need to be completed and resource acquired. Within terrorist financing there is still a flow of tasks and resources required to generate and move proceeds, whether generated illegally or legally. Any organised activity that requires multiple tasks, resources, functions etc. performed by multiple actors can be mapped and applied. This will then also guide enterprises on what data is necessary to support advanced analytics.

Furthermore, all enterprises involved in understanding the criminal domain that hope to push beyond a superficial level of analytics and risk management need to align and bundle their operational (micro and meso), strategic (macro) and strategy together into a coherent program that has high face validity for all stakeholders. This a significant extension currently in development, but out of scope within this paper.

We will now specifically outline a range of possible extensions for each of the three algorithms detailed.

Supply chain inference and identifying "Super-brokers"

Applying the notion of supply chain to the criminal domain creates useful domain context, however the level of abstraction and complexity is an important modelling decision. Of course, there needs to be a level of detail available in the data, both in terms of quality and quantity, which enables the possibility of modelling such a concept. Then the level of abstraction of supply chain needs to both be cognizant of the data available and the real-world problem. A deep understanding of the supply chain and how it operates is critical to develop this capability, both in terms of the data required and the way it is modelled. Therefore, to get into a good position to evolve the model many facets need to be addressed. These include engagement with real-world intelligence and investigations to understand what data is available, what concepts are important in combination with understanding academic empirical research and theory involving supply chain and criminal networks, and the value derived from the current applied model.

In terms of data quality and data quantity, the impact of the quality of nodes, edges, and relevant attributes remains untested at this point.

A key extension, related to data quality, is temporality. Networks evolve over time and relationships change. A person may be involved as a drug trafficker at one point in time and in another may not have any involvement in the supply chain. These elements are critical to take into account to ensure time mismatches do not create error.

Radial walk attribute propagation (RWAP)

The prediction of attitude using RWAP is at early stages of development and requires significant testing across a range of datasets that vary in scale, quantity of information available, quality of information available, and graph topology.

The model is designed to predict attitude; however attitude is yet to be explored sufficiently nor defined fully to allow for a rigorous decomposition of the core elements. This first version of RWAP is a good starting point to launch into these aspects in earnest. Of course, to forge a high value applied model a number of iterations, utilising the full value of academic research available, needs to be undertaken in an applied setting using real-world data and the real-world limitations that come along with that. Within these developmental iterations key aspects can be further explored and tested.

Experimentation on data quality and the impact of varying the quality of data such as nodes, edges, completeness, and specifically the quality of observed scores is required. Using a more rigorous approach to ensure observed scores are accurate is important. This can be done by using corroboratory information and temporal based data to better quantify the observed scores. These elements need to be framed so the generalisability of the model is retained, however knowledge is acquired in terms of the impact of data quality and potential options generated in terms of how to increase data quality.

Linked to the idea of data quality is the influence of transitivity and other topological factors on the model's performance and specifically in relation to observed scores and the graph position of these observed scores. How does RWAP performance change across different states in relation to topology? Does performance differ on nodes that are leaves, hubs, or bridges? A range of areas can be explored here and tested.

There is some early tentative evidence that optimised RWAP parameters may hold when applied across a variety of data sets. This needs to be rigorously tested. The implications of having wide generalisability of RWAP with a reasonably fixed set of parameters include the possibility that the model minimally contains some hints about criminal attitudes and direct and indirect relationships. In any case there would be significant utility in finding an association between data features and RWAP parameters as this would enable automating the optimisation of the RWAP parameters limiting the need for brute force optimisation. For example, how do the optimised parameters change when data quality decreases? A rational response may be increasing Q (the number of neighbours used to assess a neighbourhood's attitude) so that more data points are used in the prediction. Topological metrics may also be of use in understanding the relationship between performance and optimised parameters.

A whole branch of analysis can be undertaken to analyse the results of RWAP on a variety of datasets and investigate social psychology theory around behaviour and attitude in relation to networks and crime. This can then contribute to reforming the abstraction of the core problem and potential redesign of RWAP to test and potentially incorporate new elements. For example, deriving edge metrics or incorporate edge attribute information to contribute towards the information available can enhance the specificity of predictions, as not all relationships are equal. This may be applied, for example, in edge weights. In relation there is a clear opportunity to test the value of link prediction in relation to RWAP. Testing could include materialising various numbers of link predictions (and link inference) and assessing the impact, if any, on performance. This may create value in the sense of understanding the mechanisms underpinning both link prediction and RWAP, and generate applied value in both technologies.

Alternate algorithms, such as epidemiological models (e.g. SIR), random walk models, other radial walk models and agent-based models can be designed and be used to challenge the RWAP model to at least enable a more nuanced understanding of the problem, and potentially create a new model that outperforms RWAP. The process of creating a champion challenger framework may either elicit a creative solution to improve the RWAP model, generate a new higher performing model, or enable a more enhanced understanding of the problem that enables an ensemble or reinforcement learning approach.

Analysis of RWAP's value in applied settings needs to be undertaken. The iterative process of engaging with users and acquiring feedback and requirements from the real-world will boost the maturation process, quickly identifying elements that provide real-world value and those that do not. An example of this could be that that the squared difference metrics, used to measure and optimise performance, may not be so applicable in some circumstances. Perhaps a slight alteration of this metric is required to improve real-world utility. Through doing this we will get an enhanced understanding how to measure the utility of a model's output, and how many and how accurate the observed scores need to be before the model's efficacy has real-world applicability.

GraphExtract

At this point in its development "GraphExtract" requires applied maturation. In other words, "GraphExtract" needs to be applied to a variety of real-world settings to elicit performance feedback, and to understand what elements work in what circumstances and what do not. Testing on varying datasets of varying quality and completeness and contexts with differing user requirements (e.g. a national intelligence perspective) using robust experimental design will ensure wide applicability. It is only through this process that a more robust version can be developed. A range of specific potential extensions follow.

The development of a more nuanced approach to the detection of sub-types of subgraphs has the potential to generate substantial value. This can be developed in a range of ways. In situations where data is available, criminal and money-laundering modus operandi can be mapped in an appropriate abstraction (perhaps using an ontology). Other options include the development of subgraph and mesoscopic level metrics and using this enhanced knowledge to enhance the "GraphExtract" algorithm (for example, measuring the overlap between subgraphs or topological assessment). This infers a learning iterative process (e.g. reinforcement learning) which will no doubt provide enhanced results. This may come at a cost of computational expense; however, this expense may be able to be mitigated by fundamentally changing the algorithm from a serial to parallelised approach.

More specifically, experimenting with differing mechanisms to identify "entities of interest" (step 1 of the "GraphExtract" algorithm) will either uncover a superior approach, or at least enhance our understanding of how to approach this step of the algorithm. A node classification approach could augment the current approach, utilising local and global graph features in conjunction with attributes.

Within step 2 (partition the entities of interest and include mediating entities) partitioning is a core part, and therefore continuing the experimentation with both non-over-lapping and over-lapping community detection algorithms, component-based partitioning and stochastic blockmodeling to explore the value of notions of equivalence to generate a partitioned weighted distance graph. A secondary element would be generating attributes for this partitioned weighted distance graph to create a more contextual representation that allows a more nuanced set of decision-making. This may include utilising a subtler approach (e.g. RAI) than simply using graph distance to construct the weighted graph.

The detection of mediating nodes via identifying nodes on the shortest paths between each partition's "entities of interest" is an approach that is used to locate other relevant latent entities that may be of interest. This approach can be developed to include other ways to identify relevant latent entities of interest. One such extension could be including the notion of "Super-broker" or other concept and measure the value of such a change.

Within step 3 (locate boundaries of subgraphs and generate subgraphs) a core aspect is determining the boundaries of each subgraph through radial walks. As not all neighbours are equal a key focus could be continuing experimentation with random walks and vertex and edge attributes/features. In this way the identification of adjacent entities of interest and patterns of interest (e.g. using topological and temporality features) could prove useful. The detection of cycles is one such aspect, extending the path-based approach to include long cycles. The rationale for improved cycle detection is that long cycles are a very important topological feature indicating potential convoluted concealment of money. Additionally, using an ontological approach to support inference at a more

abstract level (e.g. Russian organised crime exploitation of the Cypriot banking system) would be useful to make the most out of the incomplete data that is available.

Identification of non-trust relationships and supernodes in a more enhanced, yet generalised way, will pervasively improve performance, both in terms of speeding up runtime and the value of the subgraphs extracted.

The presentation of the output from "GraphExtract" is a key element that is out of scope of this paper, and yet at some stage in the future needs to be brought into scope. The importance includes creating the ability for users to edit graph visualisations and provide feedback mechanisms enabling users to generate ideas on new features and elements that work in a sub-optimal fashion.

In terms of measurement of performance "GraphExtract" needs to be deployed over time at scale to measure the long-term value. The commitment needs to be multiyear as users require a period to understand and become efficient at extracting maximal value from such an approach.

Having outlined all the above possible extensions it is important to note that the intention of the "GraphExtract" algorithm was to detect fragments of criminal groups and activity. Many of the extensions outlined enhance the possibility of achieving this and take the idea beyond generalizable unsupervised learning into a more data-dependent unsupervised learning approach which relies on richer quality and quantity of data. The balancing act here is retaining the algorithms generalisability so it remains an open option to the widest audience, and yet takes advantage of explicit knowledge if available.

More generally

Technology such as graph partitioning, path-based subgraph extraction, and topology are used throughout the entire computational approach. A significant testing regime can be continued to understand the performance of all these functions, and assess their utility at both a component, module and system level. For example, the application of overlapping community detection approaches such as clique percolation (Palla et al., 2005) and LinkComm (Ahn et al., 2010) can be investigated further and applied in the "GraphExtract" algorithm.

Scalability again is an important element, as a lack of scalability severely limits domain applicability. Testing to date indicates a relatively short run-time for the Discover Knowledge section in the context of the overall solution with testing on the Fused Data graph of ~9 million nodes and ~ 90 million edges taking ~30 seconds for the Supply chain component, ~60 seconds for attitude prediction component and 140 minutes for the "GraphExtract" component. Theoretically the section should be

close to linear in terms of its applicability to larger graphs, dependent on what topology methods are deployed.

A number of additional extensions can be made to either make the output richer and more domain centric (e.g. providing unexplained wealth metrics in the tax domain) or generically provide value with, for example, network analytics including micro metrics; degree, betweenness, eigenvector centrality, closeness, constraint, transitivity, notions of structural equivalence through blockmodeling, bridging index, Boundary Spanners, etc., and meso/macro metrics; clustering coefficient, small-world quotient, approximate scale-free degree distribution, centralisation, cycle detection, etc. All these metrics are available as part of the GCND but have not been covered in any detail due to their non-novel nature.

This group of metrics can be used to support evolving modelling opportunities from a data perspective and / or a theoretical / empirical basis. For example, Morselli (2009) proposes that low risk appetite is associated to group topology characterised by a decentralisation and the creation of peripheral "cushions". A second example is that those entities within criminal groups that control information flow are likely to be most influential (Turner, 1991) and it is likely that the same entities that seek out control are also more likely to manifest extreme behaviours (Hare, 1999), potentially leading to group polarisation (Myers & Lamm, 1976; Taube, 2004). These types of theoretical and empirically derived hypotheses taken from both academia and from enterprise subject matter experts, can be implemented and tested. However, again it is critical to point out that all knowledge discovery endeavours rely on quality data, and particularly those more nuanced and complex constructs. This again underlines why advanced analytics that are attempting to discover latent non-linear patterns within the data absolutely require a strong focus on entity resolution and link prediction.

7.5 Other Technology Directions

At the highest level of abstraction the way in which the developed modules and sub-modules are deployed could derive significant benefit. For example, ER and LP focus on different concepts but as a by-product ER at times identifies unobserved edges between two real-world entities and LP identifies instances of where multiple entities in the data relate to a single real-world entity. Utilising the ER and LP technology in concert understanding further in the context of the problem how each technology can contribute to a better result should be explored.

Technology such as federated search can leverage off the data assets derived to generate a valuable search mechanism, bringing the fundamental value of the fused data directly to the user. Another

perspective can be utilising a combination of federated search, ER, and LP to generate real-time entity resolution predictions.

The GCND can also naturally be extended into a more macro space developing a range of useful applied strategic concepts (a strategic menu of concepts) on top of the data represented as a complex system. So high level concepts such vulnerability, sophistication, resilience, uncertainty, role, and influence, all from a systems perspective, can not only guide decisions but also ensure decisions are reflective of strategy. This would also enable the ability to drill down through strategic metrics into meso concepts right through to the micro level, utilising a rage of concepts and associated metrics to materialise latent knowledge in a variety of useful ways. This view couples strategy and the strategic view with the operational arm of organisations that focus on meso and micro levels.

The transparency of uncertainty is critical to good evidence-based decision-making. There is an opportunity to build on the uncertainty measured in a modular sense within GCND and create a more end to end approach so clarity can be provided in terms of how much certainty exists when making a decision, and the source of that uncertainty. In this way enterprise decision-making can be matured giving an explicit basis from which to deploy resource in an optimised way. Whether that resource deployment is based on understanding the problems better, acquiring better data, building better models to create better quality data, build better risk models, visualise the knowledge in more effective ways, improve governance, or in how the enterprise operationalises this knowledge.

Summary [chapter 8]

The purpose of modelling is to build a simplified representation of the real-world which enables the subsequent discovery of latent knowledge. The problem, the knowledge you want to derive, and resources available, including time, constrain the scope of how complex the model should be. Too simple and the model may not reflect reality sufficiently to enable useful insights. Too complex and the model will require considerably more resource and may not realise any more useful insights than the over-simplified model. A common mistake of the rational isolated actor approach is to limit the model to a single dataset that only provides a window to one artificially constructed component of the problem. Deriving data-centric models in this way is often driven by the inability of modellers to integrate or fuse heterogeneous datasets, who then rationalise their inherent limitations as being pragmatic. This is not pragmatism it is just poor modelling, and it is conceptually ignoring the real-world problem and instead focusing on ranking rows of data on a spreadsheet. Law enforcement over the last 20 or 30 years has demonstrated that some value can be generated from ranking spreadsheets and trying to find "risk" in data, however there has been a notable evolution in computational and criminological fields that allows problem-focussed models that are not hindered by the inability to integrate data. It stands to reason that the over-simplified data-centric approach will only enable the detection of a superficial class of criminality, whilst a more problem-centric modelling approach that more accurately reflects the real-world creates the opportunity to detect more sophisticated instances of crime and understand crime at a more nuanced level. This enhanced understanding of crime then creates the opportunity to develop a more efficacious and sustainable strategy to retard crime.

Reshaping the solution to focus on understanding and mitigating crime as a problem-centric abstraction, in combination with generating a fused dataset that represents core conceptual elements of the problem, naturally leads us away from the rational isolated actor perspective and towards the complex system paradigm. The complex system paradigm assumes that relationships count. Not only do relationships count they in fact are the basis for many criminal system features that constrain opportunities (e.g. supply chain) and influence behaviour (e.g. outlaw motorcycle gang members influencing behavioural norms). These features lead to group behaviour, natural dynamic boundaries and pathways that can manifest as emergent properties – manifestations of patterns and structure that are exhibited in higher abstract levels of the system (e.g. pathways, group or whole) that are not directly attributable to the linear aggregation of the atomic sub-components.

The opportunities that the complex system paradigm provides when trying to understand criminality has been clearly outlined through the survey of relevant theoretical and empirical findings, including those findings that have been applied to the criminal domain to date. For instance, the importance of brokers and influential entities, and weak ties in generating access to resources are all relevant

features of criminal networks at a micro level. Furthermore, graph features such as scale-free degree distribution, assortativity and network resilience enable an advanced strategy to reduce the efficiency of the criminal network in a sustainable and efficient way.

An emphasis was placed on reviewing a range of criminal domain focussed computational solutions that applied a graph concept. This provided the opportunity to highlight the differences between solutions that are query based and reactive (i.e. a real entity is required to seed the search) versus proactive applications that utilise an abstraction of the data to detect latent knowledge. This is important because law enforcement and intelligence agencies are rooted in a historically based reactive methodology that fundamentally fails to take advantage of more complex abstractions, and the potential value generated by a proactive capability. So, by detailing the differences between reactive and proactive methods, the underpinning dependencies (e.g. data, computational resource, human capability) and the value potentially generated by each approach these agencies can make an informed decision in terms of whether to invest in proactive models.

I also covered a variety of published supervised and unsupervised graph-based models applied to the criminal domain. Supervised models targeting narrow specific known problems and unsupervised models coupled to data and a high level of data quality were discussed, with an emphasis on the open world assumption. The open world assumption is a critical point of difference between data-centric and problem-centric approaches. The data-centric approach is inherently closed world as the problem is represented within the data in its entirety. Whereas the problem-centric approach is inherently open world as the data is considered a mere representation of partial elements of the real-world problem, or in other words the real-world problem will always encompass more than the data can ever represent. Therefore, any decision that is made in the open world must incorporate the assumption that there is always unobserved data. Again, the subtle distinctions between supervised and unsupervised models and the closed and open world assumption are important for evolving beyond a superficial analytical approach.

In response to the relatively unexplored opportunity to apply a complex systems approach to understand and detect criminal risk the GCND computational solution was developed. GCND is comprised of two sections which each has multiple modules. The first section is “Make Data Exploitable”. This section is comprised of two key technologies; entity resolution and link prediction. The second section is “Discover Knowledge”. This section is comprised of generating latent knowledge using a building block mentality. A battery of graph metrics, like partitioning using community detection and degree centrality, are generated in combination with contextual domain knowledge like “Super-broker”, attitude prediction, to create the opportunity to detect crime across

the micro meso macro spectrum using “GraphExtract” - enabling a complex systems view of the problem.

We aim to make data exploitable by first applying entity resolution and link prediction technology. Assuming we are presented with multiple datasets that represent core elements of the criminal problem we then need to fuse this set of heterogeneous datasets. The key technology to fuse these datasets is entity resolution. As discussed, entity resolution is focussed on the detection of real-world duplicate entities. Identifying these real-world duplicates enables the union of disparate datasets and goes beyond the relational solution of primary keys. The integration of multiple datasets unlocks our ability to better model the problem rather than attempting to rank crime using a single source of data. The EntityResolution package is built specifically as a solution to this more complex problem converse to the markets generic products that are computationally fast but inaccurate, as per the performance evaluation. The EntityResolution package is designed to conduct pairwise comparison and collective entity resolution to make contextual decisions on whether pairs of entities are in fact the same real-world entity or not. A number of novel features improve performance including utilising graph features, apply high certainty ER predictions iteratively via vertex contraction to improve the data quality, Proper Name Classifier (PNC), Proper Name Origin Classifier (PNOC), Reference Graph Algorithm (RGA), contextual collective ER, applying in situ ER, providing advanced performance metrics (i.e. global transitivity) and enable easier manual evaluation using the visualisation of a sample of the most uncertain entity resolution decisions. The performance of the EntityResolution package compared favourably with competitors with a mean F-measure of 0.9863 in comparison to the competitors 0.8723, at a slightly slower runtime. The EntityResolution package has proven scalability to graphs of ~18 million nodes and ~93 million edges, with a runtime of approximately 29 hours to completely fuse the four evaluation datasets into a single fused graph.

The LinkDiscovery package focuses on improving the quality of data, so often a big stumbling block in the analysis of criminality. The package uses link inference and an iterating radial SVM approach using a set of engineered features including Resource Allocation Index, assortativity (such as Name Origin), name similarity, age similarity, and other network metrics to predict missing links. The Link Prediction model is designed to predict both strong and weak ties to enable a more generic application in the criminal domain. The evaluation of the model’s performance is encouraging with the model successfully predicting a significant number of links across each dataset with a consistent accuracy of around 0.78 – 0.84. Of these predicted links a significant proportion are the highly valued weak ties that generate latent knowledge. This compares favourably against the most comparable approach applied by Rhodes and Jones (2009) who generated an accuracy of between 0.26 and 0.45 in the assessment of 136 pairs in a terrorist group. The computational efficiency and scalability of the Link Prediction model developed indicates utility in graphs up to a size of ~ 9 million nodes and ~ 90

million edges with a runtime of around two hours, performance that is far from prohibitive for many criminal domain applications and compares well to other approaches.

The EntityResolution and LinkDiscovery packages form the basis of the "Make Data Exploitable" section. The high functioning of this section is requisite for the successful deployment of the KnowledgeDiscovery package. This is because the upstream uncertainty that exists will only gain more momentum in knowledge discovery as the data and derivations of the data are utilised to build a series of models that are co-dependent.

The KnowledgeDiscovery package focuses on providing a range of standard entity level graph metrics in combination the "Super-broker" and RWAP attitude prediction metrics, in addition to the "GraphExtract" algorithm which detects subgraphs of criminal activity. The "Super-broker" metric utilises the fused graph to detect and assign supply chain roles and then simply count the number of times each entity directly mediates a relationship between a pair of entities that are not otherwise directly connected to someone within an adjacent role class (e.g. Trafficker → "Super-broker" → Wholesaler). The radial walk attribute prediction metric (RWAP) uses a parallel radial walk approach to propagate observed attitude scores through the network, importantly utilising community attitude to seed node labels. The "Super-broker" metrics performance is yet to be ascertained comprehensively through objective means, however the high face validity coupled to the theoretic and empirical basis for the metric indicates high value. The runtime of 27 seconds in the context of a 30+ hour total runtime means the cost is negligible. RWAP however has demonstrated high performance in predicting individuals attitude given a very small fragment of information (1.16% and 0.07% of nodes had labels (i.e. attitude scores)). We demonstrated a correlation of 0.59 and 0.78 in predicting a neighbours (N_1) attitude, and 0.44 and 0.14 in the neighbours of neighbours (N_2). N_1 predictions enabled predictions for 3.3% and 0.12% of the total number of persons, with N_2 predictions enabled predictions for a further 2.8% and 0.04% of the total number of persons. Again, RWAP has a strong theory basis and demonstrated empirical analogues.

The KnowledgeDiscovery package also includes the "GraphExtract" algorithm. This algorithm uses an abstraction of the supply chain to identify persons of interest, which are then partitioned. These partitions (with mediating nodes) are then used as the seeds for detecting subgraph boundaries using a radial walk. Each induced subgraph is then generated to form a set of subgraphs – which represent fragments of functional groups of criminal actors/events. This set of subgraphs is then used to generate a weighted mesoscopic graph with each subgraph represented as a node and each edge the proportion of nodes that exist in the intersect between each pair of subgraphs. Testing was conducted on a sample of 100 of the 20,000 subgraphs extracted from the ~9 million node 90 million edge original fused graph, with a runtime of 140 minutes. Two subject matter experts from a law

enforcement agency agreed that 90% of the subgraphs were relevant. The mesoscopic graph was also analysed by the subject matter experts who agreed that the representation and findings met their broad expectations of what the real-world criminal system looks like. We however need to consider that this visibility has not been made available prior to this work and so their views are based completely on anecdotal evidence and an assessment of plausibility. Blockmodeling was conducted on the mesograph which identified a clear core periphery structure with key brokering subgraphs detected, which again reflected what the subject matter experts expected.

The creation of a fused graph naturally creates the opportunity to go beyond a rational actor perspective and view entities as actors that operate within a complex system.

GCND creates multiple outputs, or data assets, that are of value. These are as follows:

A persisted highly accurate fused graph that contains a set of precomputed vertex attributes such as community, degree, "Super-broker" metric, and attitude. This fused graph represents entity resolution predictions in either a contracted or linked manner, dependent on user preference. In both representations the entity resolution predictions are expressed within the graph as a probabilistic machine learning prediction, making the uncertainty in each prediction explicit to the end user. A table of metadata in relation to the predictions is also generated to enable complete transparency and potential for analytical extensions (e.g. build a deep learning model to challenge the out of the box SVM) as required.

This fused graph creates the opportunity for standard entity-based query and knowledge discovery. Entity-based query, for example, could include conducting link analysis and simple data visualisation. An investigator in the border context may be interested in a specific entity and want to understand what companies this entity is associated to and what aliases are known. This can be achieved very easily because the relevant datasets have been fused and link prediction has identified aliases used by the entity. Knowledge discovery could be applied in a myriad of ways. The fused graph could be used as a source of existing features (e.g. RWAP), or as a source to engineer new features (e.g. persons that have a shareholder or director relationship with a company structure that has a presence in a tax haven jurisdiction). Another direction could be using more abstract pattern-based approaches to understand the criminal system; for example, identifying all cycles that involve suspicious transactions AND entities domiciled in a tax haven. The opportunity to build supervised and unsupervised models of this data can create advanced model output to create contextual and material risk-based 'leads' for intelligence analysts or investigators to develop.

The set of subgraphs extracted is the second primary data asset generated. This set of subgraphs can be utilised in similar ways to the original fused graph, with the advantages of having precomputed

boundaries. Each subgraph can therefore be computationally treated as a discrete entity, with a range of basic metrics pre-computed such as, number of nodes, whether organised crime entities are present, what specific transaction patterns have been identified, and what phases of the supply chain are represented. Furthermore, unsupervised learning can be applied to identify which subgraphs are similar to be clustered, giving insight into what patterns are important and how we can use this notion of subgraph similarity to infer knowledge about a subgraph where there is an absence of data. These subgraphs are also used as a contextual lead, where they are visualised and presented to the appropriate intelligence/investigative resource whom can quickly validate whether the prima facie risk is material or not and take appropriate action. For example, an investigator that is interested in a specific gang portfolio can review the relevant set of subgraphs. Alternatively, a project focusing on a specific money laundering modus operandi may be interested in all subgraphs that involve instances where domestic companies have been utilised as conduit vehicles for offshore entities.

The mesoscopic graph gives an alternate view of the criminal system. This perspective enables the application of a range of knowledge discovery technologies, creating the opportunity for significant extensions in an agencies understanding of how the criminal system operates and how to maximally inhibit its growth. For example, topological vulnerability is a well-researched concept that has high face validity when applied to the criminal domain. Another example is attempting to identify phase transitions where a functional group is going through a transitional change from a position of strength to a position of weakness (e.g. a gang has lost members) and therefore may be vulnerable to intervention. A third example is understanding how the supply chain operates, how functional groups fit, and identifying what operational strategies could be applied to maximally disrupt this supply chain, with a view of measuring the impact and learning. A whole range of structural and pattern-based approaches become viable applied options in terms of gaining an understanding of the criminal system.

At the macro level the data assets outputted from GCND can be used in concert to develop sets of descriptive analytics, such as the number of criminal convictions conducted by what level of grouping over what time period. However, the opportunity also presents to explore more advanced ways of gaining insight into the system. Changes in the ways classes of individuals and groups are influencing the performance of the entire network are of significant interest. This could be concrete where an understanding of how accountants and lawyers are complicit or actively facilitate criminal activity, or more abstract where the focus is a more generic notion of brokers and how they are either complicit or actively facilitate criminal activity.

The important point to make here is that because of the creation of high-quality fused data assets that express slightly different views of the same criminal system there is the opportunity to develop

understanding in a variety of ways that can support all elements of informed decision-making across an enterprise.

Whether that decision is at the entity level where the data assets are used as a reference database to augment current investigations or to generate entity leads. At the meso level where better decisions can be made in terms of how to deploy operational resource at what areas of the criminal system in what way and in what order. For example, the development of a coherent proactive plan in terms of where to focus data collection efforts over a multi-year plan in conjunction with what intelligence and investigation resource to deploy in what capacity. At a macro strategy level, the new knowledge available can inform strategy decisions, internal capability required, what data sets to invest into, what computational assets to invest into (infrastructure, capability models, application layer), and what governance structures to invest into to ensure efficient risk management is achieved in a consistent and internally transparent way.

Criminality is not an individualistic phenomenon. Criminality is a phenomenon, like any other human behaviour, that is founded in social interaction. Whether it is the influence of others, the resource supplied by others, or the normalisation of attitudes provided by others, it is clear that there is an inter-dependence between criminal actors. Therefore, the problem must be constructed in terms of actors comprising an interdependent system rather than as a set of isolated rational actors. The only way to model such a problem is by taking these connections between entities into account, which will require the integration of multiple relevant heterogeneous datasets that represent core elements of the real-world criminal problem. Entity resolution provides the basis to fuse data and link prediction generates higher quality data and reducing incompleteness and uncertainty. Subsequent to representing the data in an expressive and exploitable graph knowledge discovery can then derive contextual entity level value by identifying the most significant actors, and in parallel provide group level value by creating visibility over how functional groups of actors inter-relate within the overarching criminal system and give context in terms of how best to create a strategy to best depress the functioning of this criminal system. Making this leap will firmly move law enforcement and intelligence agencies from the unsustainable easy “low hanging fruit” approach to the more productive efficient and effective approach that focuses on “high hanging fruit”.

References

1. Abadinsky, H. (2012). *Organized Crime (10 edition)*. Wadsworth, OH: Cengage Learning.
2. Adamic, L., & Adar, E. (2003). Friends and Neighbors on the Web. *Social Networks*, *25*, 211–230.
3. Adler, P. A. (1985). Wheeling and dealing. *An Ethnography of an Upper-Level Drug Dealing and Smuggling Community*. New York.
4. Ahn, Y. Y., Bagrow, J. P., & Lehmann, S. (2010). Link Communities Reveal Multiscale Complexity in Networks. *Nature*, *466*(7307), 761-764. https://doi.org/10.1038/nature09182
5. Albanese, J. S. (2012). The Cosa Nostra in the US Adapting to Changes in the Social, Economic, and Political Environment After a 25-Year Prosecution Effort. In D. Siegel & H. van de Bunt (Eds.), *Traditional Organized Crime in the Modern World* (pp. 93–108). Springer. Retrieved from http://link.springer.com/chapter/10.1007/978-1-4614-3212-8_5
6. Albert, R., Jeong, H., & Barabási, A.-L. (2000). Error and attack tolerance of complex networks. *Nature*, *406*(6794), 378–382.
7. Albert, R., & Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, *74*(1), 47–97. https://doi.org/10.1103/RevModPhys.74.47
8. Allaire, J. J., Xie, Y., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., Wickham, H., Cheng, J., Chang, W., & Iannone, R. (2019). rmarkdown: Dynamic Documents for R. R package version 1.16. https://rmarkdown.rstudio.com. Accessed 9 Dec 2019.
9. Almende, B.V., Thieurmel, B., & Robert, T. (2019). visNetwork: Network Visualization using 'vis.js' Library. R package version 2.0.8. https://CRAN.R-project.org/package=visNetwork. Accessed 9 Dec 2019.
10. Amadore, N. (2007). La zona grigia. *Professionisti Al Servizio*.
11. Amaral, L. A. N., Scala, A., Barthélémy, M., & Stanley, H. E. (2000). Classes of small-world networks. *Proceedings of the National Academy of Sciences*, *97*(21), 11149–11152. https://doi.org/10.1073/pnas.200327197
12. Asch, S. E. (1951). Effects of group pressure upon the modification and distortion of judgments. In *Groups, leadership and men; research in human relations* (pp. 177–190). Oxford, England: Carnegie Press.
13. Bakker, R. M., Raab, J., & Milward, H. B. (2012). A preliminary theory of dark network resilience. *Journal of Policy Analysis and Management*, *31*(1), 33–62.
14. Bandura, A. (1971). *Social learning theory*. New York: General Learning Press.
15. Barabási, A.-L. & Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, *286*, 509–512. https://doi.org/10.1126/science.286.5439.509

16. Barabási, A.-L. (2009). Scale-Free Networks: A Decade and Beyond. *Science*, *325*(5939), 412–413. https://doi.org/10.1126/science.1173299

17. Beken, T. V. (2004). Risky business: A risk-based methodology to measure organized crime. *Crime, Law and Social Change*, *41*(5), 471–516. https://doi.org/10.1023/B:CRIS.0000039599.73924.af

18. Benjelloun, O., Garcia-Molina, H., Menestrina, D., Su, Q., Whang, S. E., & Widom, J. (2009). Swoosh: A Generic Approach to Entity Resolution. *The VLDB Journal*, *18*(1), 255-276. https://doi.org/10.1007/s00778-008-0098-x

19. Benson, J. S., & Decker, S. H. (2010). The organizational structure of international drug smuggling. *Journal of Criminal Justice*, *38*(2), 130–138.

20. Berlusconi, G., Calderoni, F., Parolini, N., Verani, M., & Piccardi, C. (2016). Link Prediction in Criminal Networks: A Tool for Criminal Intelligence Analysis. *PLoS ONE 11*(4): e0154244. https://doi.org/10.1371/journal.pone.0154244

21. Bhargava, A., & Kondrak, G. (2010). Language identification of names with SVMs. Conference: Human Language Technologies: *Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 2-4, 2010*, Los Angeles, California, USA. 693-696.

22. Bhattacharya, I., & Getoor, L. (2006). Entity Resolution in Graphs. In: Cook, D. J. & Holder, L. B. (eds.) *Mining Graph Data*, pp. 311–344. John Wiley & Sons, Inc., Hoboken, NJ, USA.

23. Bhattacharya, I., & Getoor, L. (2007). Collective Entity Resolution in Relational Data. *ACM Transactions on Knowledge Discovery from Data 1*, 1, 1-36. https://doi.org/10.1145/1217299.1217304

24. Bichler, G., & Malm, A. E. (2015). Why Networks? In G. Bichler, & A. E. Malm (Eds.), *Disrupting Criminal Networks: Network Analysis in Crime Prevention* (pp. 1-8). First Forum Press.

25. Bilgic, M., Licamele, L., Getoor, L., & Shneiderman, B. (2006). D-dupe: An interactive tool for entity resolution in social networks. In *2006 IEEE Symposium on Visual Analytics Science and Technology* (pp. 43–50). IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4035746

26. Black, C., & Beken, T. V. (2001). *Reporting on Organised Crime: A Shift from Description to Explanation in the Belgian Annual Report on Organised Crime*. Maklu.

27. Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. (2008). Fast Unfolding of Communities in Large Networks. *Journal of Statistical Mechanics: Theory and Experiment. vol 2008*, 10, P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008

28. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., & Hwang, D.-U. (2006). Complex networks: Structure and dynamics. *Physics Reports*, *424*(4–5), 175–308. https://doi.org/10.1016/j.physrep.2005.10.009

29. Boivin, R. (2013). Drug trafficking networks in the world-economy. In C. Morselli (Ed), *Crime and networks* (pp. 182-191). United States: Taylor and Francis. https://doi.org/10.4324/9781315885018
30. Bonacich, P. (1987). Power and Centrality: A Family of Measures. *American Journal of Sociology*, *92*(5), 1170–1182.
31. Borgatti, S. P. (2003). The Key Player Problem1. *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers*, *241*.
32. Borgatti, S. P. (2005). Centrality and network flow. *Social Networks*, *27*(1), 55–71.
33. Borgatti, S. P. (2006). Identifying sets of key players in a social network. *Computational & Mathematical Organization Theory*, *12*(1), 21–34. https://doi.org/10.1007/s10588-006-7084-x
34. Borgatti, S. P., & Everett, M. G. (1992). Notions of position in social network analysis. *Sociological Methodology*, *22*(1), 1–35.
35. Borgatti, S. P., & Everett, M. G. (2006). A graph-theoretic framework for classifying centrality measures. *Social Networks 28*(4): 466-484. https://doi.org/10.1016/j.socnet.2005.11.005
36. Brantingham, P. L., Ester, M., Frank, R., Glässer, U., & Tayebi, M. A. (2011). *Co-offending network mining*. Springer.
37. Bright, D. A., & Delaney, J. J. (2013). Evolution of a drug trafficking network: Mapping changes in network structure and function across time. *Global Crime*, *14*(2-3), 238–260.
38. Bright, D. A., Greenhill, C., & Levenkova, N. (2011). Dismantling criminal networks: can node attributes play a role. In *Illicit Networks Conference* (pp. 1–29).
39. Bright, D. A., Hughes, C. E., & Chalmers, J. (2012). Illuminating dark networks: a social network analysis of an Australian drug trafficking syndicate. *Crime, Law and Social Change*, *57*(2), 151–176.
40. Broido, A. D., & Clauset, A. (2018). Scale-free networks are rare. arXiv:1801.03400 [physics.soc-ph]
41. Burt, R. S. (2004). *Brokerage and Closure: An Introduction to Social Capital*. Clarendon Lectures in Management Studies.
42. Calderoni, F. (2011). Strategic positioning in mafia networks. In *Third Annual Illicit Networks Workshop*, Montreal.
43. Calderoni, F. (2015). Predicting Organized Crime Leaders. In G. Bichler, & A. E. Malm (Eds.), *Disrupting Criminal Networks: Network Analysis in Crime Prevention* (pp. 89-110). First Forum Press.
44. Callaway, D. S., Newman, M. E. J., Strogatz, S. H., & Watts, D. J. (2000). Network Robustness and Fragility: Percolation on Random Graphs. *Physical Review Letters*, *85*(25), 5468–5471.
45. Carley, K. M., 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

46. Carley, K. M. (2003). Dynamic network analysis. In R. Breiger, K. M. Carley, & P. Pattison (Eds.), *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers* (pp. 133–145). Committee on Human Factors, National Research Council, Washington DC.
47. Carley, K. M. (2006). Destabilization of covert networks. *Computational & Mathematical Organization Theory, 12*(1), 51–66.
48. Carley, K. M., Lee, J.-S., & Krackhardt, D. (2002). Destabilizing Networks. *Connections, 24*(3), 79-92.
49. Castells, M. (1996). *The Rise of the Network Society: The Information Age: Economy, Society, and Culture*. Chichester, West Sussex ; Malden, MA: Wiley-Blackwell.
50. Chen, P. P.-S. (1976). The Entity-relationship Model—Toward a Unified View of Data. *ACM Trans. Database Syst., 1*(1), 9–36. https://doi.org/10.1145/320434.320440
51. Chen, H., Chung, W., Xu, J. J., Wang, G., Qin, Y., & Chau, M. (2004). Crime data mining: a general framework and some examples. *Computer, 37*(4):50–56.
52. Christakis, N. A., & Fowler, J. H. (2007). The Spread of Obesity in a Large Social Network over 32 Years. *New England Journal of Medicine, 357*(4), 370–379.
53. Clauset, A., Moore, C., & Newman, M. E. J. (2008). Hierarchical structure and the prediction of missing links in networks. *Nature, 453*(7191), 98–101.
54. Clauset, A., Newman, M. E. J., & Moore, C. (2004). Finding community structure in very large networks. *Physical Review E, 70*(6).
55. Cleckley, H. M. (1988). *The Mask of Sanity: An Attempt to Clarify Some Issues About the So Called Psychopathic Personality (5th edition)*. Augusta, Georgia: Emily S. Cleckley.
56. Coles, N. (2001). It's Not What You Know—It's Who You Know That Counts. Analysing Serious Crime Groups as Social Networks. *British Journal of Criminology, 41*(4), 580–594.
57. Csardi, G., & Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal Complex Syst*. http://igraph.org. Accessed 6 Dec 2017.
58. Davis, J. A. (1979). The Davis/Holland/Leinhardt studies: An overview. In P. W. Holland & S. Leinhardt (Eds.), *Perspectives on Social Network Research* (pp. 51-62). Academic Press. https://doi.org/10.1016/B978-0-12-352550-5.50009-2
59. Desroches, F. J. (2005). *The crime that pays: Drug trafficking and organized crime in Canada*. Canadian Scholars' Press.
60. Doreian, P., Ferligoj, A., Batagelj, V., & Granovetter, M. (2005). *Generalized Blockmodeling*. Cambridge University Press: New York, NY, USA.
61. Dorn, N., Murji, K., & South, N. (1992). *Traffickers: Drug Markets and Law Enforcement*. London: Routledge.

62. Dowle, M., & Srinivasan, A. (2019). data.table: Extension of `data.frame`. R package version 1.12.4. https://CRAN.R-project.org/package=data.table. Accessed 9 Dec 2019.
63. Duijn, P. A. C., & Klerks, P. P. H. M. (2014). Social Network Analysis Applied to Criminal Networks: Recent Developments in Dutch Law Enforcement. In A. Masys (Ed.), *Networks and Network Analysis for Defence and Security. Lecture Notes in Social Networks* (pp. 121–159). Springer, Cham. https://doi.org/10.1007/978-3-319-04147-6_6
64. Dunbar, R. I. M. (1992). Neocortex size as a constraint on group size in primates. *J Hum Evol 22*(6):469–493.
65. Dunn, H. L. (1946). Record Linkage. *American Journal of Public Health and the Nations Health*, *36*(12), 1412–1416.
66. Europol. (2013). *Serious and Organised Crime Threat Assessment*. The Hague: Europol.
67. Everett, M. G., & Borgatti, S. P. (1999). The centrality of groups and classes. *The Journal of Mathematical Sociology*, *23*(3), 181–201.
68. Everton, D. S. F. (2013). *Disrupting Dark Networks*. New York, NY: Cambridge University Press.
69. Faloutsos, M., Faloutsos, P., & Faloutsos, C. (1999). On Power-law Relationships of the Internet Topology. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication* (pp. 251–262). New York, NY, USA: ACM.
70. FATF/OECD. (2010). *Money Laundering Using Trust and Company Service Providers*. Paris: FATF/OECD.
71. Faust, K. (1988). Comparison of methods for positional analysis: Structural and general equivalences. *Social Networks*, *10*(4), 313–341.
72. Faust, K., & Wasserman, S. (1992). Blockmodels: Interpretation and evaluation. *Social Networks*, *14*(1–2), 5–61. https://doi.org/10.1016/0378-8733(92)90013-W
73. Feld, S. L. (1981). The Focused Organization of Social Ties. *American Journal of Sociology*, *86*(5), 1015–1035.
74. Fellegi, I. P., & Sunter, A. B. (1969). A Theory for Record Linkage. *Journal of the American Statistical Association*, *64*(328), 1183–1210.
75. Festinger, L. (1956). *When prophecy fails*. University of Minnesota Press.
76. Fire M., Puzis R., & Elovici Y. (2013). Link Prediction in Highly Fractional Data Sets. In V. Subrahmanian (Ed.), *Handbook of Computational Approaches to Counterterrorism* (pp. 283-300). Springer, New York, NY. https://doi.org/10.1007/978-1-4614-5311-6_14
77. Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, *486*(3-5), 75–174. https://doi.org/10.1016/j.physrep.2009.11.002

78. Fortunato, S., & Barthélemy, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences of the United States of America*, *104*(1), 36–41. https://doi.org/10.1073/pnas.0605965104
79. Freeman, L. C. (1977). A Set of Measures of Centrality Based on Betweenness. *Sociometry*, *40*(1), 35–41.
80. Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks*, *1*(3), 215–239.
81. French, J. R. P., & Raven, B. (1959). The bases of social power. In D. Cartwright & A. Zander (Eds.), *Group dynamics* (pp. 151-164). New York: Harper & Row.
82. Fu, Y., Xu, F., & Uszkoreit. (2010). Determining the Origin and Structure of Person Names. *LREC*.
83. Gagolewski, M. (2019). R package stringi: Character string processing facilities. http://www.gagolewski.com/software/stringi/. Accessed 9 Dec 2019.
84. Galaskiewicz, J. (1979). The structure of community organizational networks. *Social Forces*, *57*(4), 1346–1364. https://doi.org/10.1093/sf/57.4.1346
85. Galeotti, M. (2012). Turkish Organised Crime: From Tradition to Business. In D. Siegel & H. van de Bunt (Eds.), *Traditional Organized Crime in the Modern World* (pp. 49–64). Springer.
86. Ghoshal, G., & Barabási, A.-L. (2011). Ranking stability and super-stable nodes in complex networks. *Nature Communications*, *2*(394). https://doi.org/10.1038/ncomms1396
87. Gimenez-Salinas Framis, A. (2014). Illegal networks or criminal organizations: Structure, power, and facilitators in cocaine trafficking structures. In C. Morselli (Ed.), *Crime and Networks* (pp. 131–147). New York, NY: Routledge.
88. Giuffre, K. (2013). *Communities and Networks: Using Social Network Analysis to Rethink Urban and Community Studies*. John Wiley & Sons.
89. Goldstein, R. (1999). Emergence as a construct: History and issues. *Emergence*, *1*(1), 49–72.
90. Gould, R. V. (1989). Power and social structure in community elites. *Social Forces*, *68*(2), 531–552. https://doi.org/10.2307/2579259
91. Gould, R. V., & Fernandez, R. M. (1989). Structures of mediation: A formal approach to brokerage in transaction networks. *Sociological Methodology*, *19*(1989), 89–126. https://doi.org/10.2307/270949
92. Granovetter, M. S. (1973). The strength of weak ties. *The American Journal of Sociology*, *78*(6), 1360-1380.
93. Guimerà, R., & Sales-Pardo, M. (2009). Missing and spurious interactions and the reconstruction of complex networks. *Proceedings of the National Academy of Sciences of the United States of America*, *106*(52), 22073–22078.
94. Hare, R. D. (1999). *The Hare Psychopathy Checklist-Revised: PLC-R*. MHS, Multi-Health Systems.

95. Harper, W. R., & Harris, D. H. (1975). The application of link analysis to police intelligence. *Hum Factors 17*(2):157–164.
96. Hasan, M. A., Chaoji, V., Salem, S., & Zaki, M. (2006). Link Prediction using Supervised Learning. In *Proceedings of SDM 06 workshop on Link Analysis, Counterterrorism and Security*.
97. Heider, F (1958). *The Psychology of Interpersonal Relations*. John Wiley & Sons.
98. Hernández, M. A., & Stolfo, S. J. (1995). The Merge/Purge Problem for Large Databases. In *Proceedings of the ACM SIGMOD International Conference on Management of Data 1995*, pp. 127–138.
99. Hernández, M. A., & Stolfo, S. J. (1998). Real-world Data is Dirty: Data Cleansing and The Merge/Purge Problem. *Data Min. Knowl. Discov.*, *2*(1), 9–37. https://doi.org/10.1023/A:1009761603038
100. Hernando, A., Villuendas, D., Vesperinas, C., Abad, M., & Plastino, A. (2009). Unravelling the size distribution of social groups with information theory on complex networks. *Eur. Phys. J. B*, *76*(1), 87-97. https://doi.org/10.1140/epjb/e2010-00216-1
101. Holland, P. W., & Leinhardt, S. (1971). Transitivity in Structural Models of Small Groups. *Small Group Research*, *2*(2), 107–124.
102. Holme, P., Kim, B. J., Yoon, C. N., & Han, S. K. (2002). Attack vulnerability of complex networks. *Physical Review E*, *65*(5), 056109.
103. Hsu, T.-W., Wu, C. W., Cheng, Y.-F., Chen, H.-L., Lu, C.-H., Cho, K.-H., & Lin, C.-P. (2012). Impaired Small-World Network Efficiency and Dynamic Functional Distribution in Patients with Cirrhosis. *PLoS ONE*, *7*(5), e35266.
104. Hu, D., Kaza, S., & Chen, H. (2009). Identifying significant facilitators of dark network evolution. *Journal of the American Society for Information Science and Technology*, *60*(4), 655–665.
105. Huang, D., Mu, D., Yang, L., & Cai, X. (2018). CoDetect: financial fraud detection with anomaly feature detection. *IEEE Access 6*, 19161–19174. https://doi.org/10.1109/ACCESS.2018.2816564
106. Jacobs, J. B., & Peters, E. (2003). Labor racketeering: The mafia and the unions. *Crime and Justice*, *30*(2003), 229–282.
107. Jaro, M. A. (1989). Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. *Journal of the American Statistical Association, 84*(406), 414–420.
108. Jaro, M. A. (1995). Probabilistic linkage of large public health data files. *Statistics in Medicine, 14*(5-7), 491–498.

109. Jeh, G., & Widom, J. (2002). SimRank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 538–543). ACM. https://doi.org/10.1145/775107.775126

110. Jenkins, P., & Potter, G. (1987). The politics and mythology of organized crime: a Philadelphia case-study. *Journal of Criminal Justice, 15*(6), 473–484.

111. Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N., & Barabási, A.-L. (2000). The large-scale organization of metabolic networks. *Nature, 407*(6804), 651–654.

112. Jones, J. H., & Handcock, M. S. (2003). An assessment of preferential attachment as a mechanism for human sexual network formation. *Proceedings of the Royal Society of London B: Biological Sciences, 270*(1520), 1123–1128.

113. 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). https://doi.org/10.1137/1.9781611972870.13

114. Karatzoglou, A., Smola, A., Hornik, K., & Zeileis, A. (2004). kernlab - An S4 Package for Kernel Methods in R. *Journal of Statistical Software 11*(9), 1-20. http://www.jstatsoft.org/v11/i09/. Accessed 9 Dec 2019.

115. Katz, L. (1953). A new status index derived from sociometric analysis. *Psychometrika, 18*(1), 39–43. https://doi.org/10.1007/BF02289026

116. Kelman, H. C. (1961). Processes of opinion change. *Public Opinion Quarterly, 25*(1), 57–78.

117. Kim, M., & Leskovec, J. (2011). The Network Completion Problem: Inferring Missing Nodes and Edges in Networks. In *Proceedings of the Eleventh SIAM International Conference on Data Mining, SDM 2011, April 28-30, 2011, Mesa, Arizona, USA* (pp. 47–58). SIAM / Omnipress. https://doi.org/10.1137/1.9781611972818.5

118. 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.

119. Klik, M. (2019). fst: Lightning Fast Serialization of Data Frames for R. R package version 0.9.0. https://CRAN.R-project.org/package=fst. Accessed 9 Dec 2019.

120. Kossinets, G., & Watts, D. J. (2006). Empirical Analysis of an Evolving Social Network. *Science, 311*(5757), 88–90.

121. Krebs, V. E. (2002). Mapping networks of terrorist cells. *Connections, 24*(3), 43–52.

122. Krogan, N. J., Cagney, G., Yu, H., Zhong, G., Guo, X., Ignatchenko, A., & Greenblatt, J. F. (2006). Global landscape of protein complexes in the yeast Saccharomyces cerevisiae. *Nature, 440*(7084), 637–643.

123. Lambert, D. M., & Cooper, M. C. (2000). Issues in Supply Chain Management. *Industrial Marketing Management, 29*, 65–83.

124. Latapy, M., Magnien, C., & Vecchio, N. D. (2008). Basic Notions for the Analysis of Large two-mode Networks. *Social Networks*, *30*(1), 31-48. https://doi.org/10.1016/j.socnet.2007.04.006

125. Lee, E.-J. (2007). Deindividuation Effects on Group Polarization in Computer-Mediated Communication: The Role of Group Identification, Public-Self-Awareness, and Perceived Argument Quality. *Journal of Communication*, *57*(2), 385–403.

126. Leuprecht, C., & Hall, K. (2014). Why Terror Networks are Dissimilar: How Structure Relates to Function. In A. Masys (Ed.), *Networks and Network Analysis for Defence and Security. Lecture Notes in Social Networks* (pp. 83-120). Springer, Cham. https://doi.org/10.1007/978-3-319-04147-6_6

127. Li, J., & Wang G, A. (2013). Criminal Identity Resolution Using Personal and Social Identity Attributes: A Collective Resolution Approach. In C. Yang, W. Mao, X. Zheng, & H. Wang (Eds.), *Intelligent Systems for Security Informatics* (pp. 107-124). Boston, Academic Press. https://doi.org/10.1016/B978-0-12-404702-0.00006-9

128. Liben-Nowell, D., & Kleinberg, J. (2007). The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, *58*(7), 1019–1031.

129. 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).

130. Li, F., He, J., Huang, G., Zhang, Y., & Shi, Y. (2014). A Clustering-based Link Prediction Method in Social Networks. *Procedia Computer Science*, *29*, 432–442.

131. Lim, E.-P., Srivastava, J., Prabhakar, S., & Richardson, J. (1993). Entity Identification in Database Integration. In *Proceedings of the Ninth International Conference on Data Engineering* (pp. 294–301). Washington, DC, USA: IEEE Computer Society.

132. Lisbach, B., & Meyer, V. (2013). *Linguistic Identity Matching*. Wiesbaden: Springer Fachmedien Wiesbaden.

133. Lorrain, F., & White, H. C. (1971). Structural equivalence of individuals in social networks. *The Journal of Mathematical Sociology*, *1*(1), 49–80.

134. Lo, T. W., & Kwok, S. I. (2012). Traditional organized crime in the modern world: how triad societies respond to socioeconomic change. In D. Siegel & H. van de Bunt (Eds.), *Traditional Organized Crime in the Modern World* (pp. 67-89). Springer.

135. Lü, L., Pan, L., Zhou, T., Zhang, Y. C., & Stanley, H. E. (2015). Toward link predictability of complex networks. *Proceedings of the National Academy of Sciences of the United States of America*. *112*(8):2325–2330. https://doi.org/10.1073/pnas.1424644112

136. Lu, L., & Zhou, T. (2009). Role of Weak Ties in Link Prediction of Complex Networks. In *Proceedings of the First ACM International Workshop on Complex Networks Meet Information & Knowledge Management, NY, USA* (pp. 55–58).

137. Luce, R. D., & Perry, A. D. (1949). A method of matrix analysis of group structure. *Psychometrika*, *14*(2), 95–116. https://doi.org/10.1007/BF02289146
138. Maeno, Y. (2009). Node Discovery Problem for a Social Network. *Connections. 29*, 62-76.
139. Malm, A., & Bichler, G. (2011). Networks of Collaborating Criminals: Assessing the Structural Vulnerability of Drug Markets. *Journal of Research in Crime and Delinquency*, *48*(2), 271–297.
140. Malm, A., Bichler, G., & Nash, R. (2011). Co-offending between criminal enterprise groups. *Global Crime*, *12*(2), 112–128. https://doi.org/10.1080/17440572.2011.567832
141. Mandel, M. J. (1983). Local roles and social networks. *American Sociological Review*, *48*(3), 376–386. https://doi.org/10.2307/2095229
142. Marsden, P. V. (1982). Brokerage behavior in restricted exchange networks. *Social Structure and Network Analysis*, *7*(4), 341–410.
143. Martínez, V., Berzal, F., & Cubero, J-C. (2016). A Survey of Link Prediction in Complex Networks. *ACM Comput. Surv.* *49*(4), Article 69 (December 2016). https://doi.org/10.1145/3012704
144. Maydanchik, A. (2007). *Data Quality Assessment*. USA: Technics Publications, LLC.
145. McAndrew, D. (1999). The structural analysis of criminal networks. In D. Canter & L. Alison (Eds.), *The social psychology of crime: Groups, teams, and networks, offender profiling series, iii*, (pp. 51-94). Aldershot: Dartmouth.
146. McCallum, A., Nigam, K., & Ungar, L. H. (2000). Efficient clustering of high-dimensional data sets with application to reference matching. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 169–178). ACM. https://doi.org/10.1145/347090.347123
147. McGloin, J. M., & Nguyen, H. (2013). The importance of studying co-offending networks for criminological theory and policy. In C. Morselli (Ed.), *Crime and networks* (pp. 13-27). United States: Taylor and Francis. https://doi.org/10.4324/9781315885018
148. McIllwain, J. S. (1999). Organized crime: A social network approach. *Crime, Law and Social Change*, *32*(4), 301–323. https://doi.org/10.1023/A:1008354713842
149. McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology*, *27*(1), 415–444.
150. Merton, R. K. (1968). *Social theory and social structure*. Simon and Schuster.
151. 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).
152. Microsoft Corporation & Weston, S. (2019). doParallel: Foreach Parallel Adaptor for the 'parallel' Package. R package version 1.0.15. https://CRAN.R-project.org/package=doParallel. Accessed 9 Dec 2019.

153. Microsoft Corporation & Weston, S. (2019). foreach: Provides Foreach Looping Construct. R package version 1.4.7. https://CRAN.R-project.org/package=foreach. Accessed 9 Dec 2019.
154. Milgram, S. (1963). Behavioral study of obedience. *Journal of Abnormal and Social Psychology Today, 67*(4), 371-378.
155. Milgram, S. (1967). The small world problem. *Psychology Today, 2*(1), 60–67.
156. Monge, A., & Elkan, C. (1996). The field matching problem: Algorithms and applications. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (pp. 267–270).
157. Montgomery, R. (1976). Outlaw Motorcycle Subculture. *Canadian Journal of Criminology and Corrections, 18,* 332-342.
158. 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).
159. Morrison, S., & Australian Institute of Criminology. (2002). *Approaching organised crime: where are we now and where are we going?* Canberra: Australian Institute of Criminology.
160. Morselli, C. (2003). Career opportunities and network-based privileges in the Cosa Nostra. *Crime, Law and Social Change, 39*(4), 383–418.
161. Morselli, C. (2005). *Contacts, opportunities, and criminal enterprise*. Toronto ; Buffalo: University of Toronto Press.
162. Morselli C. (2009). *Inside criminal networks*. New York: Springer.
163. Morselli, C. (2010). Assessing vulnerable and strategic positions in a criminal network. *Journal of Contemporary Criminal Justice, 26*(4), 382–392.
164. Morselli, C., & Giguere, C. (2006). Legitimate strengths in criminal networks. *Crime, Law and Social Change, 45*(3), 185–200.
165. Morselli, C., Grund, T., & Boivin, R. (2015). Network stability issues in a co-offending population. In G. Bichler, & A. E. Malm (Eds.), *Disrupting Criminal Networks: Network Analysis in Crime Prevention* (pp. 47-65). First Forum Press.
166. Morselli, C., & Petit, K. (2007). Law-enforcement disruption of a drug importation network. *Global Crime, 8*(2), 109–130.
167. Morselli, C., & Roy, J. (2008). Brokerage Qualifications in Ringing Operations. *Criminology, 46*(1), 71–98. https://doi.org/10.1111/j.1745-9125.2008.00103.x
168. Morselli, C., & Tremblay, P. (2004). Criminal achievement, offender networks and the benefits of low self-control. *Criminology, 42*(3), 773–804. https://doi.org/10.1111/j.1745-9125.2004.tb00536.x
169. Morselli, C., Tremblay, P., & McCarthy, B. (2006). Mentors and criminal achievement. *Criminology, 44*(1), 17–43. https://doi.org/10.1111/j.1745-9125.2006.00041.x

170. Myers, D. G., & Lamm, H. (1976). The group polarization phenomenon. *Psychological Bulletin, 83*(4), 602-627. http://dx.doi.org/10.1037/0033-2909.83.4.602

171. Murata, T., & Moriyasu, S. (2007). Link Prediction of Social Networks Based on Weighted Proximity Measures. *IEEE/WIC/ACM International Conference on Web Intelligence (WI'07)* (pp. 85-88). Fremont, CA. https://doi.org/10.1109/WI.2007.52

172. Nadeau, D., & Sekine, S. (2007). A survey of named entity recognition and classification. In S. Sekine & E. Ranchhod (Eds.), *Named Entities: Recognition, classification and use [Special issue of Lingvisticæ Investigationes 30:1]* (pp. 3-26). John Benjamins Publishing Company. https://doi.org/10.1075/li.30.1.03nad

173. Natarajan, M. (2000). Understanding the structure of a drug trafficking organization: a conversational analysis. *Crime Prevention Studies, 11*, 273–298.

174. Natarajan, M. (2006). Understanding the structure of a large heroin distribution network: A quantitative analysis of qualitative data. *Journal of Quantitative Criminology, 22*(2), 171–192.

175. Natarajan, M., & Belanger, M. (1998). Varieties of drug trafficking organizations: a typology of cases prosecuted in New York City. *Journal of Drug Issues, 28*(4), 1005.

176. Naumann, F., & Herschel, M. (2010). An Introduction to Duplicate Detection. *Synthesis Lectures on Data Management*, *2*(1), 1–87. https://doi.org/10.2200/S00262ED1V01Y201003DTM003

177. Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, *48*(3), 443–453.

178. Newcomb, T. M. (1953). An approach to the study of communicative acts. *Psychological Review, 60*, 393-404.

179. Newcombe, H. B., & Kennedy, J. M. (1962). Record Linkage: Making Maximum Use of the Discriminating Power of Identifying Information. *Commun. ACM, 5*(11), 563–566.

180. Newcombe, H. B., Kennedy, J. M., Axford, S. J., & James, A. P. (1959). Automatic Linkage of Vital Records Computers can be used to extract “follow-up” statistics of families from files of routine records. *Science, 130*(3381), 954–959.

181. Newman, M. E. J. (2001). Clustering and preferential attachment in growing networks. *Physical Review E, 64*(2), 025102. https://doi.org/10.1103/PhysRevE.64.025102

182. Newman, M. (2003). The Structure and Function of Complex Networks. *SIAM Review, 45*(2), 167–256. https://doi.org/10.1137/S003614450342480

183. Newman, M. E. J. (2003). Mixing patterns in networks. *Physical Review E, 67*(2) 026126. https://doi.org/10.1103/PhysRevE.67.026126

184. Newman, M. (2004). Fast algorithm for detecting community structure in networks. *Physical Review E, 69*(6), 066133. https://doi.org/10.1103/PhysRevE.69.066133

185. Newman, M. E., & Girvan, M. (2003). Finding and evaluating community structure in networks. *Physical Review E*, *69*(2), 026113. https://doi.org/10.1103/PhysRevE.69.026113
186. Nobesawa, S. & Tahara, I. (2005). Language Identification for Person Names Based on Statistical Information. In *Proceedings of the 19th Pacific Asia Conference on Language, Information and Computation* (pp. 289-296). Institute of Linguistics, Academia Sinica.
187. Odell, M., & Russell, R. (1918). *The Soundex Coding System*. US Patents 1261167.
188. OFCANZ. (2010). *Organised Crime in New Zealand*. Wellington: OFCANZ.
189. Ozgul, F., Erdem, Z., Bowerman, C., & Bondy, J. (2010). Combined detection model for criminal network detection. In H. Chen, M. Chau, S. Li, S. Urs, S. Srinivasa, & G. A. Wang (Eds.), *Intelligence and security informatics. Lecture notes in computer science* (pp. 1-14). Springer, Berlin. https://doi.org/10.1007/978-3-642-13601-6_1
190. Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). The PageRank citation ranking: bringing order to the Web. *Technical Report 1999-66*, Stanford InfoLab.
191. 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.
192. Paoli, L. (2002). The paradoxes of organized crime. *Crime, Law and Social Change*, *37*(1), 51–97.
193. Philips, L. (2002). The Double Metaphone Search Algorithm, *C/C++ Users Journal 18*(6), 38-43.
194. de sola Pool, I., & Kochen, M. (1978). Contacts and influence. *Social Networks*, *1*(1), 5–51.
195. Powers, D. M. W. (2011). Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies*. *2*(1), 37–63.
196. Prado, A., Plantevit, M., Robardet, C., & Boulicaut, J. F. (2013). Mining graph topological patterns: finding covariations among vertex descriptors. *IEEE Trans Knowl Data Eng 25*(9), 2090–2104.
197. Quinn, J. F. (2001). Angels, bandidos, outlaws, and pagans: The evolution of organized crime among the big four 1% motorcycle clubs. *Deviant Behavior*, *22*(4), 379–399.
198. R Core Team. (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/. Accessed 9 Dec 2019.
199. Raghavan, U. N., Albert, R., & Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E*, *76*(3), 036106. https://doi.org/10.1103/PhysRevE.76.036106
200. Randall, S. M., Boyd, J. H., Ferrante, A. M., Bauer, J. K., & Semmens, J. B. (2014). Use of graph theory measures to identify errors in record linkage. *Computer Methods and Programs in Biomedicine*, *115*(2), 55–63. https://doi.org/10.1016/j.cmpb.2014.03.008

201. Ratcliffe, J., Strang, S., & Taylor, R. (2014). Assessing the success factors of organized crime groups: Intelligence challenges for strategic thinking. *Policing: An International Journal of Police Strategies & Management*, *37*(1), 206–227.
202. Ravasz, E., Somera, A. L., Mongru, D. A., Oltvai, Z. N., & Barabási, A. L. (2002). Hierarchical organization of modularity in metabolic networks. *Science*, *297*(5586), 1551–1555. https://doi.org/10.1126/science.1073374
203. Reiss Jr, A. J. (1988). Co-offending and criminal careers. *Crime and Justice*, *10*, 117–170.
204. Reuter, P. H., & Haaga, J. (1989). *The organization of high-level drug markets*. Santa Monica, CA: Rand.
205. Revesz, P. Z. (1993). On the semantics of theory change: Arbitration between old and new information. In *Proceedings of the twelfth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems* (pp. 71–82). ACM.
206. Rhodes, C. J., & Jones, P. (2009). Inferring missing links in partially observed social networks. *Journal of the Operational Research Society*, *60*(10), 1373–1383. https://doi.org/10.1057/jors.2008.110
207. Robinson, D. (2016). The Use of Reference Graphs in the Entity Resolution of Criminal Networks. In M. Chau, G. A. Wang, & H. Chen (Eds.), *PAISI 2016. LNCS, 9650* (pp. 3-18). Springer, Cham. https://doi.org/10.1007/978-3-319-31863-9_1
208. Robinson, D., & Scogings, C. (2017). Picking High Level Fruit in Dark Trees: Using Complex Systems Analytics to Detect and Understand Crime. In A. Colarik, J. Jang-Jaccard, & A. Mathrani (Eds.), *Cyber Security and Policy: A substantive dialogue* (pp. 87-108). Auckland: Massey University Press.
209. Robinson, D., & Scogings, C. (2018). The detection of criminal groups in real-world fused data: using the graph-mining algorithm "GraphExtract". *Security Informatics*, *7*(2), 1. https://doi.org/10.1186/s13388-018-0031-9
210. Rodríguez, J. A. (2005). The March 11th Terrorist Network: In its weakness lies its strength. In *Proceedings XXV International Sunbelt Conference*, Los Angeles.
211. Rosvall, M., & Bergstrom, C. T. (2008). Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, *105*(4), 1118–1123.
212. Savona, E. U. (2012). Italian mafias' asymmetries. In D. Siegel & H. van de Bunt (Eds.), *Traditional Organized Crime in the Modern World* (pp. 3-25). Springer.
213. Schelling, T. C. (1969). *Models of segregation*. Rand Corp.
214. 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.
215. Shaikh, M. A., Wang, J., Yang, Z., & Song, Y. (2007). Graph structural mining in terrorist networks. In R. Alhajj, H. Gao, J. Li, X. Li, & O. R. Zaïane (Eds.), *Advanced Data Mining and*

Applications, ADMA 2007, LNCS, 4632 (pp. 570-577). Berlin, Heidelberg: Springer. https://doi.org/10.1007/978-3-540-73871-8_54

216. Shakarian, P., Martin, M., Bertetto, J. A., Fischl, B., Hannigan, J., Hernandez G., Kenney, E., Lademan, J., Paulo, D., & Young, C. (2015). Criminal social network intelligence analysis with the gang software. In L. M. Gerdes (Ed.), *Illuminating Dark Networks* (pp. 143-156). Cambridge University Press. https://doi.org/10.1017/CBO9781316212639.010
217. Siegel, D. (2012). Vory v zakone: Russian Organized Crime. In D. Siegel & H. van de Bunt (Eds.), *Traditional Organized Crime in the Modern World* (pp. 27-47). Springer.
218. Simmel, G., & Wolff, K. H. (1950). *The sociology of georg simmel* (Vol. 92892). Simon and Schuster.
219. Smets, P. H. (1988). Belief functions. In P. H. Smets, A. Mamdani, D. Dubois, & Prade, H. (Eds.), *Non-Standard Logics for Automated Reasoning* (pp. 253–286). London: Academic Press.
220. Smith Sr, R. C., & Fox, G. W. (2002). Dangerous motorcycle gangs: A facet of organized crime in the mid-Atlantic region. *Journal of Gang Research, 9*(4), 33-44.
221. Sparrow, M. (1991). The application of network analysis to criminal intelligence: An assessment of the prospects. *Social Networks, 13*(3), 251–274.
222. Sutherland, E. H. (1937). *The Professional Thief*. Chicago: The University of Chicago press.
223. Talburt, J. R. (2011). *Entity resolution and information quality*. San Francisco, Calif: Morgan Kaufmann/Elsevier.
224. Täube, V. G. (2004). Measuring the social capital of brokerage roles. *Connections, 26*(1), 29–52.
225. Therneau, T., & Atkinson, B. (2019). rpart: Recursive Partitioning and Regression Trees. R package version 4.1-15. https://CRAN.R-project.org/package=rpart. Accessed 9 Dec 2019.
226. Thuraisingham, B. (2003). *Web Data Mining and Applications in Business Intelligence and Counter-Terrorism*. CRC Press.
227. Tretheway, S., & Katz, T. (1998). Motorcycle gangs or motorcycle mafia? *Police Chief, 65*, 53–61.
228. Tuckman, B. W. (1965). Developmental sequence in small groups. *Psychological Bulletin, 63*(6), 384.
229. Turner, J. C. (1991). *Social influence*. Milton Keynes, [England]: Open University Press.
230. Tushman, M. L. (1977). Special boundary roles in the innovation process. *Administrative Science Quarterly, 22*(4), 587–605. https://doi.org/10.2307/2392402
231. Tusikov, N. (2010). The Godfather is Dead: A Hybrid Model of Organized Crime. In G. Martinez-Zalace, S. Vargas Cervantes, & W. Straw (Eds.), *Aprehendiendo al Delincuente: Crimen y Medios en America del Norte* (pp. 143-159). Media at McGill.

232. Ugander, J., Karrer, B., Backstrom, L., & Marlow, C. (2011). The Anatomy of the Facebook Social Graph. *CoRR*, abs/1111.4503.

233. Urbanek, S. (2017). fastmatch: Fast match() function. R package version 1.1-0. https://CRAN.R-project.org/package=fastmatch. Accessed 9 Dec 2019.

234. van der Loo, M. (2014). The stringdist package for approximate string matching. The R Journal, 6, 111-122. https://CRAN.R-project.org/package=stringdist. Accessed 9 Dec 2019.

235. Van Mastrigt, S. B., & Carrington, P. (2013). Sex and Age Homophily in Co-offending Networks. In C. Morselli (Ed), *Crime and networks* (pp. 28-51). Taylor and Francis, United States. https://doi.org/10.4324/9781315885018

236. Van Rijsbergen, C. J. (1979). *Information retrieval (2d ed)*. London ; Boston: Butterworths.

237. Varese, F. (2013). The Structure and the Content of Criminal Connections: The Russian Mafia in Italy. *European Sociological Review, 29*(5), 899–909.

238. Veno, A. (2002). *The Brotherhoods: Inside the Outlaw Motorcycle Clubs*. Allen & Unwin.

239. 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.

240. Wasserman, S., & Faust, K. (1994). *Social network analysis: methods and applications.* Cambridge ; New York: Cambridge University Press.

241. Watts, D. J. (1999). Networks, Dynamics, and the Small-World Phenomenon. *American Journal of Sociology, 105*, 493–527.

242. Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of "small-world" networks. *Nature, 393*(6684), 440–442.

243. Weerman, F. M. (2003). Co-offending as Social Exchange. Explaining Characteristics of Co-offending. *British Journal of Criminology, 43*(2), 398–416.

244. Whang, S. E., Menestrina, D., Koutrika, G., Theobald, M., & Garcia-Molina, H. (2009). Entity Resolution with Iterative Blocking. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data* (pp. 219–232). New York, NY, USA: ACM.

245. White, D. R., & Harary, F. (2001). The Cohesiveness of Blocks in Social Networks: Node Connectivity and Conditional Density. *Sociological Methodology, 31*(1), 305–359.

246. White, D. R., & Reitz, K. P. (1983). Graph and semigroup homomorphisms on networks of relations. *Social Networks, 5*(2), 193–234.

247. White, H. C., Boorman, S. A., & Breiger, R. L. (1976). Social structure from multiple networks. I. Blockmodels of roles and positions. *American Journal of Sociology, 81*(4), 730–780.

248. Wickham, H. (2007). Reshaping Data with the reshape Package. *Journal of Statistical Software, 21*(12), 1-20. http://www.jstatsoft.org/v21/i12/. Accessed 9 Dec 2019.

249. Wickham, H. (2019). stringr: Simple, Consistent Wrappers for Common String Operations. R package version 1.4.0. https://CRAN.R-project.org/package=stringr. Accessed 9 Dec 2019.

250. Williams, P., & Godson, R. (2002). Anticipating organized and transnational crime. *Crime, Law and Social Change, 37*(4), 311–355. https://doi.org/10.1023/A:1016095317864

251. Winkler, W. E. (1990). String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. In *1990 Proceedings of the Section on Survey Research Methods, American Statistical Association*, 354-359.

252. Winship, C. (1988). Thoughts about roles and relations: an old document revisited. *Social Networks, 10*(3), 209–231.

253. Winship, C., & Mandel, M. (1983). Roles and positions: A critique and extension of the blockmodeling approach. *Sociological Methodology, 14*, 314–344.

254. Xie, Y. (2019). knitr: A General-Purpose Package for Dynamic Report Generation in R. R package version 1.25. Accessed 9 Dec 2019.

255. Xu, J., & Chen, H. (2003). Untangling Criminal Networks: A Case Study. In H. Chen, R. Miranda, D. D. Zeng, C. Demchak, J. Schroeder, & T. Madhusudan (Eds.), *Intelligence and Security Informatics* (pp. 232–248). Berlin Heidelberg: Springer.

256. Xu, J., & Chen, H. (2005). Criminal network analysis and visualization. *Communications of the ACM, 48*(6), 100–107.

257. Xu, J., & Chen, H. (2008). The topology of dark networks. *Communications of the ACM, 51*(10), 58.

258. Yip, A. M., & Horvath, S. (2007). Gene network interconnectedness and the generalized topological overlap measure. *BMC Bioinformatics, 8*, 22. https://doi.org/10.1186/1471-2105-8-22

259. Yongxin, Z., Qingzhong, L., & Ji, B. (2009). Enhancing collective entity resolution utilizing Quasi-Clique similarity measure. In *2009 Joint Conferences on Pervasive Computing (JCPC)* (pp. 263–266). https://doi.org/10.1109/JCPC.2009.5420180

260. Zhou, T., Lü, L., & Zhang, Y.-C. (2009). Predicting missing links via local information. *The European Physical Journal B, 71*(4), 623–630.

261. Zimbardo, P. G. (1970). The human choice: Individuation, reason, and order versus deindividuation, impulse, and chaos. In W. J. Arnold & D. Levine (Eds.), *1969 Nebraska Symposium on Motivation* (pp. 237-307). Lincoln, NE: University of Nebraska Press.

Glossary

Adamic-Adar metric; a similarity measure where the number of the pair's common neighbour's is weighted by the inverse log of the pair's degree. 23

Alias; a variant set of biographical details for a person. 13, 144

Anglicisation; the adoption of a 'local' English language given name by immigrants. 66, 93

Apache Spark; a distributed computing platform. 35

Approximate string matching (ASM); an algorithmic metric of string similarity. 16

Assortativity or assortative mixing; a preference for similar vertices to be connected. 48

Betweenness centrality; a shortest path based centrality measure. 39

Bipartite graph; a graph constructed from vertices of two classes, where no vertices of the same class are adjacent. 12

Blocking algorithms; a set of algorithms that aim to divide the set of elements into blocks, such that 'similar' pairs are contained in the same blocks. 17, 66

Blockmodeling; a method designed to partition vertices based on their equivalence. 31, 192

Boundary spanners; vertices that span multiple communities. 44

Boundedness; a measurement of how bounded the search space is. 69

Bridging Index; whether a person holds a link that bridges another community or not. 44, 135

Brokerage; a vertices mediation of the interaction between two alters. 41, 150

Centralization; a measure of how central the network's most central node is in relation to how central all the other nodes are. 50

Clique percolation algorithm; a community detection algorithm that identifies the union of adjacent k-cliques, where those k-cliques are assessed as being adjacent if they share $k-1$ nodes. 29

Closed world assumption; that the domain is contained within the data available. 17

Collective ER (CER); an entity resolution approach that looks at clusters of potential matches and uses transitivity (transitive closure) and exclusivity logic to support decision-making. 16, 109

Common neighbours; the number of neighbouring nodes shared by node i and node j. 23

Community detection; ascribe membership to vertices based on the pattern of relationships they have in relation to each other. 27

Companies Office; a register of the registration of companies. 59

Complex systems; characterised by complex inter-related facets between entities and the collective network of entities, the emergent properties that manifest from this interaction, the significant degree of randomness that is present in the system, the high degree of self-organising, and the broader context in which the problem space is nested. 2, 51

Contextualisation; the application of the domain's context to the problem. 36, 150

Cosine ASM; an approximate string matching algorithm. 66

Covert human intelligence source (CHIS); an informant. 7

Dark Network data; data of criminal entities. 58

Data cleansing; the process of improving the quality of data. 64

Data fusion; the process of integrating multiple datasets using entity resolution. 15

Data modelling; the process of transforming data into the desired data representation. 12

Deduplication; the process of finding and eliminating redundant duplicate information. 15

Degree centrality; the measurement of how many neighbours a node is connected to. 39

Diameter; the maximal distance between any pair of vertices. 18

Duplicates; two or more records relating to the same real-world entity. 15

Edit distance; a metric that records the minimum number of operations required to transform one string into another. 87

Eigenvector centrality; a centrality measure that identifies those entities that are connected to highly connected entities. 38

Emergent properties, emergence; phenomena where properties measured at the collective do not quantitively reflect those properties as a sum of properties of the individual actors. 51

Entity based data integration (EBDI); the process of integrating of multiple data sets. 15

Entity resolution; the detection and resolution of instances of multiple entities in data that refer to a single real-world entity. 15, 61

Equivalence; from a graph perspective the similarity of vertices pattern of connections. 27, 30

Exclusivity; the transitive-based notion that *i* cannot be equivalent to *j* if either *i* is not equivalent to *k* or *j* is not equivalent to *k*. 69, 111

Fast Greedy community detection; hierarchical based community detection algorithm. 28

F-measure; the weighted harmonic mean of the recall and precision of the test results. 16

Frequent subgraph mining (FSM); pattern detection on the topological structure of a subgraph, through inference of specific patterns or the identification of subgraph isomorphisms. 33

Gatekeeper broker; mediates interaction between a node within his / her community and a node outside of his / her community. 41

Gazetteer; an external or derived dataset of entity labels and associated meta-data designed to assist in computational decision-making. 19, 89

Generalised topological overlap measure (GTOM); a generalisation of the topological overlap measure that enables the neighbourhood concept to be extended by propagating through the neighbours to a fixed order (x). 24

Graph; defined as a set of vertices and a set of edges that connect vertices ($G=(V,E)$). 12

Graph-based Criminal Network Detection (GCND); A graph–based computational solution to detect latent criminal networks. 2

Graph distance (geodesic distance); the distance between a pair of vertices is the number of edges in the shortest path connecting that pair. 23

Graph mining; the detection of patterns in graphs, including techniques such as blockmodeling, frequent subgraph mining (FSM), and graph clustering. 33

GraphExtract; an algorithm designed to discover contextual latent knowledge from multiple perspectives across the micro – meso – macro spectrum. 165

Group polarisation; propensity for a collective to make more extreme decisions than each of its constituent members would outside of the collective. 52, 154

Harmonisation; the process of transforming heterogeneous data into homogeneous data so that data elements and representations are consistent across all input data sources. 64

Hypocorism; a nickname or diminutive name given to a person. 19

Identity resolution; the specific resolution of a new dataset against a reference set of known identities. 15

In situ ER prediction; the discovery of new knowledge gained from the instantiation of ER predictions onto the source data. 120

Influence, social; the broad area of study that encompasses how individual's behaviour and beliefs are modified by the social interactions that occur over time. 42

InfoMap algorithm; an algorithm by Rosvall and Bergstrom (2008) that detects communities through finding an optimal compression of the graph topology. 29

Jaccard coefficient; a vertex similarity measure that extends the common neighbour metric by taking into consideration the number of connections the pair of vertices have. 23

Jaro-Winkler string distance algorithm; an extension of the Jaro distance algorithm, used to measure string dissimilarity. 16

Knowledge discovery; the process of extracting relevant and useful knowledge from data for user consumption. 26, 148

Organised crime; a formal structured group of individuals coalescing for the purpose of engaging in a pattern of enduring serious criminal activity. 29

Outlaw motorcycle gang (OMG); an organised criminal group that are identifiable by the overt nature of their membership – exhibited by the wearing of a distinctive patch and rocker on a leather jacket and riding high powered motorcycles. 30

PageRank algorithm; an algorithm that measures the importance of a vertex based on the number and quality of the source vertices connected by incoming edges. 23

Pairwise intractability; the number of pairwise operations increases exponentially as the set size increases, creating a computational efficiency problem. 66

Panama Papers; the Panama Papers was leaked from the Panama law firm Mossack Fonseca and made available online in 2016. 58

Paradise Papers; The Paradise Papers was leaked from the offshore law firm Appleby and made available in 2017 and 2018. 58

Partitioning graphs; the process of dividing the vertices of a graph into subsets, based on the relationships present within the graph. 27, 149

Preferential attachment; the concept that those entities that substantively have more of something will in future generate more of that something in comparison to entities that substantively have less of that thing. 22

Precision; the fraction of correct predictions (TP) over all predictions (TP + FP). 57

Problem-focussed; the paradigm of focussing on the problem, ensuring the data collected is a satisfactory representation of the problem we are trying to solve. The closer the data represents the problem the more likely that analytic approaches will generate value associated to resolving that problem, or set of problems. 10

Proper name; a label that is used to identify a person, place or thing. 87

Proper Name Classifier (PNC); an algorithm designed to identify instances of where pairs of person entities may have similar names but are not likely to refer to the same real-world as they both contain proper names that are similar but differ (e.g. “Ken” and “Ben”). 87

Proper Name Origin Classifier (PNOC); a classifier, using machine learning, that predicts what the etymological origin is of a proper name. 93

Property graph; a graph representation that is comprised of vertices, edges, and properties (or attributes) of both vertices and edges. 12

Provenance; a chronology of how data has evolved from source through to current. 15

Psychopathy; a personality construct typified by anti-social behaviour and traits such as narcissism, lack of empathy, lack of remorse, and disinhibition. 42

Rational isolated actor; the approach to crime detection based on detecting criminal actors from one main data source, rather than a collection of data sources that better represent the problem. The Rational isolated actor approach assumes criminal actors act independently. 1

Recall; fraction of the correct predictions (TP) over all relevant instances (TP + FN). 57

Record linkage; the specific task of finding equivalent records within a set. 15

Recursive partitioning (RPART); a statistical method using decision trees to generate classifications and / or predictions. 69

Reference Graph Algorithm (RGA); a graph based blocking algorithm. 20, 102

Resource Allocation Index (RAI); an algorithm that measures the product of the normalised degree of all nodes along every shortest path between a pair. The RAI is useful in a range of domains including as a feature in Link Prediction models. 115, 133, 176

Sanctions data; is open data that includes individuals, corporate entities, and relevant associated entities that pose economic, trade and national security risks. 57

Scale-free degree distribution, approximate; studies have found that social networks are likely to display a degree distribution that approximately follows a power law. This means that a very small fraction of vertices (known as supernodes) have many connections and a long tail of vertices maintain a very small number of connections. 46, 76

Secondary data sources; data sources - such as gazetteers - that can augment computational solutions. 86

Simple graph; a graph that has undirected edges and no loops. 12

Small-world; a notion based on anyone in a network being able to communicate with anyone else within the network through only a very small number of intermediaries. 49

Social distance; a metric indicating whether a pair of nodes are proximate in the graph, computed through deploying community detection and neighbourhood distance (order of *k*). 68

Social network analysis (SNA); analysis of the structure of networks. 2

Strong ties; high trust and enduring relationships between people that are highly clustered and derive largely from cyclic closure. 127

Supernodes; vertices with a high number of connections (see Scale-free degree distribution, approximate). 58, 77

Super-brokers; a small set of brokers that indirectly connect a large proportion of entities engaging in the illicit drug supply chain. 150

Supply chain, illicit; a supply chain consisting of 'producing, trafficking, wholesaling, retailing, consuming, & money laundering'. 37, 150

Support vector machine (SVM); a supervised machine learning approach to classification, using hyperplanes. 97, 127

Suspicious Transactions; transactions that are suspected to be related to a criminal offence, notably money laundering, terrorism, and misuse of drugs. 58

8, 127

Appendix A - Mechanics of the Pairwise Equivalence wrapper function

Synopsis

The pairwise equivalence wrapper function is deployed within the Obvious Resolution and Non-Obvious Resolution sub-modules. The Obvious Resolution sub-module is designed to search for all available entity types [person, organisation, address, email, phone, bank account] and if detected identifies "exact" or "near exact" matches that have requisite data points (i.e. we have enough information about each pair to be very confident they refer to the same entity) and contract this pair in the graph retaining the metadata. The metadata is retained for provenance, auditability, and predictive purposes later. The Obvious Resolution sub-module is implemented to ensure the model is computational scalable. Failure to implement this tranche would otherwise result in intractability issues concerning the deduplication of supernodes. Supernodes in these contexts can follow a scale-free distribution, meaning that not only are these nodes over-represented in the data but to the extent that when we are dealing with datasets in the millions they create pairwise intractability. For example, a node that has 20,000 duplicates would generate 199,990,000 pairs.

The Non-Obvious Resolution sub-module is designed to identify close to all non-obvious pairs using node attributes and graph attributes, accepting there will be a significant portion of false positives within the table of pairs generated. It is important to note that the decision of whether the pair is a match or not is not made at this point but at the end of the solution when we have maximal metadata available.

Framework of how this is achieved

The framework consists of applying a specific wrapper function multiple times with varying parameter settings targeting various specific subsets of entities/scenarios. This is an important conceptual decision as the umbrella problem of entity resolution is actually comprised of a cluster of many different manifestations and associated causal origins. For example, duplicate pairs of entities can be derived from keying error and manifest as character transposition, derive from legal name changes and manifest as compound family names, or derive from an intent to obfuscate manifesting as a name variant. Each applied function generates a table of potential matches with supplementary metadata, dependent on parameters set, at both the pair level and the function level. The metadata from each applied wrapper function is then combined and used to contract the graph.

The wrapper function is named ER_Deduplication() and has the following arguments:

- g - the graph to apply the pairwise entity resolution
- Label Truncation Threshold – the number of characters at which to truncate the label
- Index Truncation Threshold – the number of characters the entity must have to be included in the set
- Object Type – the type of entity to focus the pairwise similarity on (e.g. "person"). Note that there are two variants for addresses. "Address" is the standard approach, and "Address_Num" is a variant that extracts the numbers from the address and generates labels based on a concatenation of the numbers and the truncated address.
- ASM threshold – the threshold used in conjunction with the ASM
- Blocking Construct – the sub-setting (or blocking) approach to use. See below for options.
- Meta-blocking Construct – the meta-sub-setting (or meta-blocking) approach to use. See below for options.
- Function Label –the label for the function

And if the Object type is "person" then the following arguments are also available:

- Indexing – the method to constrain the set of "persons". Options include "FN1N2DOBi", "FN1N2N3i", "FN1N2i", and "FN1i" in order of reducing the constraint. FN represents the entity must contain a family name, N1 the first given name, etc., and DOB the date of birth.
- Labelling – the method to constrain what attributes the labels are generated from. Options include "FN1N2DOBl", "FN1N2N3l", "FN1N2l", and "FN1l".
- Name Origin – the option to include the Proper Name Origin Classifier meta-blocking technique. Options currently include "FALSE" to not include (default), "CN" (Chinese name origin), "AR" (Arabic name origin), "LA" (Latin name origin), "SC" (Indian sub-continent name origin), "RU" (Slavic name origin), and "Other" (the remainder of names).
- Method – the ASM algorithm used. Options currently include "Jaro-Winkler" and "Cosine".
- Name23 – Boolean argument of whether to eliminate pairs where the second or third given names are significantly different. Currently applied at threshold of 0.965 using the Jaro-Winkler ASM.
- Name1 - Boolean argument of whether to eliminate pairs where the first given name is significantly different. Currently applied at threshold of 0.965 using the Jaro-Winkler ASM.
- Neighbourhood distance – a numeric argument (default = 0). The union of the source node and the target nodes neighbourhood, with the order determined by the numeric argument, for each pair.
- Graph distance – Boolean argument. Determines whether the pair are members of the same community, using the Louvain method.

- Name Frequency Threshold – a numeric argument [0-1]. This argument determines the threshold to apply to exclude pairs that have a name frequency higher than the threshold.
- Hypocorism – Boolean argument (default = FALSE). This argument constrains the index to include only those pairs that contain a name present in the Hypocorism Graph, and then measures name attributes via ASM as a tuple.
- DOB Difference – a numeric argument [0-1]. This argument determines how similar the DOB of the pair needs to be to be accepted. Age, day of birth, month of birth, and year of birth elements are considered.
- Graph Output – Boolean output. Whether to generate a graphml output of the results enabling manual verification.

Blocking Constructs available:

Truncation methods include “First Letter”, “First Second Letter” and “First Sixth Ninth Letter” which truncate the label or family name, if a “person” entity type, after the first letter, the first and second letter, and the first, sixth, and ninth letter respectively to create blocking “keys”. For example, the keys generated for the “person” entity “FABRICIO ALTAMIRANO” are “A”, “AL”, and “AIN” respectively.

Phonetic methods include “Soundex” and “Metaphone 3”, which use a phonetic approach to determine keys. For example, the keys generated for the “person” entity “FABRICIO ALTAMIRANO” are “A435” and “ALTM” respectively.

The community method – “CD” – uses the Louvain method to detect communities of the set within the graph and assigns “keys” based on this membership. This method is subject to error dependent on the topology of the target graph. If the topology displays topological features of a very small world and extreme scale-free degree distribution then communities may be large and not useful, so the context of when this blocking construct is deployed is very important.

Reference Graph methods include “FNRG” and “Hypocorism Graph”. The Family Name Reference Graph (FNRG) uses the communities generated from the FNRG (see above) and the Hypocorism Graph uses the component membership as a simple method of determining blocks.

Meta-blocking Constructs available:

The only meta-blocking construct available is name-origin. So, this is limited to person entities. The Name origin meta-blocking algorithm generates classes limited to the following origin classes:

Arabic, Chinese, Japanese, Latin, Persian, Russian, Sub-continent, West African, Other, and any combination of the classes (for example, the name "John Wong" would have the class "Other_Chinese" as "John" falls under the given name class Other and "Wong" under the family name class Chinese).

The wrapper function itself works by running through three steps

Step 1 – Pre-Indexing

This step identifies the set of entities to compare and generates labels for comparison. The set is first identified by the object type and then a series of exclusions are made to reduce the set to ensure accuracy. For example, any addresses that do not have a number and contain only one word (e.g. "Auckland") is not included within the set to compare. Two outputs are generated:

- Index
- Labels

Step 2 – Equivalence Assessment

This step takes the index and labels as inputs and conducts pairwise assessment of the labels. Equivalence Assessment is conducted in one of three ways dependent on the function arguments supplied and the size of the set under assessment.

If the ASM threshold is under one, or in other words not an exact match, then a specific sub-function is deployed to assess similarity, based on the arguments given. The blocking construct is organised hierarchically so if a specific method is selected and the resultant blocks are too large (i.e. the largest blocking set is over a specific threshold) to efficiently compute then a more granular blocking construct is deployed. This ensures that the wrapper function can be used by lay-persons and operate successfully. The output generated is in the form of a graph and table output of the pairs, their ASM similarity, and parameter settings.

If the ASM threshold is one, or in other words an exact match, and the set size is under a specific threshold then no blocking is undertaken and a graph and table output of the pairs and their ASM similarity are directly generated. This option optimises the entity resolution of small datasets.

If the ASM threshold is one, or in other words an exact match, and the set size is over a specific threshold then blocking is undertaken (as per the construct method selected) and a graph and table output of the pairs and their ASM similarity are directly generated. This option optimises the entity resolution of small datasets. When the ASM threshold is set at 1 the function does not deploy an

ASM algorithm to compute the pairwise similarity but deploys a Hash table using coarse grained parallelism making use of the cores available.

Step 3 – Decision Management

This step takes the graph generated from the previous step and conducts a range of additional computational operations (see arguments) to the pair to ascertain similarity, and uses this metadata to exclude pairs.

Additionally, the specific functions performance is then measured by counting the number of pairs generated and the global transitivity of those pairs.

The wrapper functions output is a list containing:

- The ASM threshold used [numeric vector]
- The number of pairs generated [integer vector]
- The global transitivity of the pairs generated [numeric vector]
- The ASM distance of each pair [numeric vector]
- The runtime of each step in the function [numeric vector in seconds]
- The edgelist of pairs generated [2 column matrix].

The wrapper functions are deployed in the Non-Obvious Resolution sub-module as follows in table 2:

Table 2. This table lists the key wrapper argument settings within Non-Obvious Resolution sub-module.

Entity Type	Name Origin	ASM Threshold	Method	Negation	Blocking Construct	Index | Label
Address	NA	1	NA	NA	CD	Address_Number
Person	CN	0.98	cosine	False	FNRG - N	FN1i | FN1N2l
Person	CN	0.98	jaro-winkler	True	FNRG	FN1i | FN1l
Person	CN	0.95	jaro-winkler	True	Hypocorism	FN1i | FN1l
Person	CN	0.96	jaro-winkler	True	FNRG	FN1DOBi | FN1N2N3DOBl
Person	CN	0.98	cosine	True	FNRG	FN1N2i | FN1N2l
Person	CN	0.96	jaro-winkler	True	FNRG	FN1DOBi | FN1DOBl
Person	CN	0.97	jaro-winkler	False	FNRG	FN1DOBi | FN2N1DOBl
Person	CN	0.98	jaro-winkler	False	FNRG - N	FN1i | FN1N2N3DOBl
Person	Other	0.96	cosine	False	FNRG - N	FN1i | FN1N2l
Person	Other	0.95	jaro-winkler	True	FNRG	FN1i | FN1l
Person	Other	0.87	jaro-winkler	True	Hypocorism	FN1i | FN1l
Person	Other	0.89	jaro-winkler	True	FNRG	FN1DOBi | FN1N2N3DOBl
Person	Other	0.92	cosine	True	FNRG	FN1N2i | FN1N2l
Person	Other	0.91	jaro-winkler	True	FNRG	FN1DOBi | FN1DOBl
Person	Other	0.90	jaro-winkler	False	FNRG	FN1DOBi | FN2N1DOBl
Person	Other	0.92	jaro-winkler	False	FNRG - N	FN1i | FN1N2N3DOBl

Table 2. displays the key wrapper argument default settings within tranche 2. The address function is designed to make use of the contracted graph by using the CD blocking construct and a reduced label

truncation point concatenated with the address numbers (e.g. the label for the entity "8 DEAN RYLE ST APARTMENT 1010 LONDON SW1P4DA UK" is "8101014DEANRYLESTA"). So, only matching addresses that are in the same community will be identified as a potential match. Note that with the wrapper functions that focus on "person" entities chunk one focusing on names of Chinese origin, due to the linguistic difference have a much more conservative ASM threshold. Also, the Cosine ASM method is restrictively used by always allying it with name negation or using some social distance measure (such as common neighbourhood). This is due to the fact the algorithm does not penalise for the order of characters and therefore any pair of names that contains the same set of letters will score 1 – an exact match. The Cosine algorithm is used specifically to identify pairs that have the same name however the order of names is different due to transcription or other error. This problem is much more common in the Chinese name origin class. The Family Name Reference Graph (FNRG) algorithm is almost exclusively used for blocking due to its highly accurate performance (see Robinson, 2016). The FNRG-N variant refers to using the FNRG for blocking in conjunction with the Neighbourhood distance feature (order=2). The Hypocorism graph is also used to specifically target entities using nicknames or diminutives.

A table of output, which we will call the "Prediction Data", is generated from all the deployed wrapper functions outlined above within tranche 2 and combined with the "person" targeted pairs generated in Tranche 1. This table of pairs and associated metadata (including ASM string distance, Name Frequency, Local Edge Transitivity, Social distance, and Information Quantity) is then used as the input to the Collective Equivalence Resolution sub-module.

Appendix B - Ethical considerations

The ethical considerations of this research were explored with the Massey University ethics committee providing the following response to the research outline:

> Friday, 10 November 2017 1:11 PM
>
> Hello Chris, thanks for sharing this dilemma. In my view the data being accessed is publically available so does not require permission for its use, but there is the potential for the analysis of data from the different sources to create an apparently damaging representation of an individual or company and this raises issues of natural justice and ethics. My impression is that the data is being used to 'prove' the capability of the software. But if the names of actual entities are to be used in the dissertation and/or any dissemination then I think that an ethics application would be required and would include explanation and justification about how the publically available 'personal' information sets were to be handled after analysis.
>
> Happy to talk further about this,
>
> Regards,
>
> Brian
>
> **Dr Brian Finch**
>
> **Director, Research Ethics**

See 3.1 Evaluation Methodology on page 54 for details of how ethical risk was mitigated.

Appendix C - DRC16

DRC 1b

MASSEY UNIVERSITY
GRADUATE RESEARCH SCHOOL

STATEMENT OF CONTRIBUTION
DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the candidate and the candidate's Primary Supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the candidate's contribution as indicated below in the *Statement of Originality*.

Name of candidate:	David Robinson
Name/title of Primary Supervisor:	Chris Scoglings
Name of Research Output and its References:	
In which Chapter is the Manuscript /Published work	NA
Please indicate:	
• The percentage of the manuscript/Published Work that was contributed by the candidate:	100%
and	
• Describe the contribution that the candidate has made to the Manuscript/Published Work:	
Completed every aspect of Robinson, D. (2016). The Use of Reference Graphs in the Entity Resolution of Criminal Networks. In M. Chau, G. A. Wang, & H. Chen (Eds.), PAISI 2016. LNCS, 9650 (pp. 3-18). Springer, Cham. https://doi.org/10.1007/978-3-319-31863-9_1	
For manuscripts intended for publication please indicate target journal:	
Candidate's Signature:	
Date:	16/05/2019
Primary Supervisor's Signature:	
Date:	20/05/2019

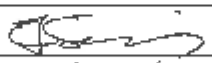
(This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis)

GRS Version 4 - January 2019

MASSEY UNIVERSITY
GRADUATE RESEARCH SCHOOL

STATEMENT OF CONTRIBUTION
DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the candidate and the candidate's Primary Supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the candidate's contribution as indicated below in the *Statement of Originality*.

Name of candidate:	David Robinson
Name/title of Primary Supervisor:	Chris Scogings
Name of Research Output and full reference:	
[illegible]	
In which Chapter is the Manuscript /Published work	Chapter 4
Please Indicate:	
• The percentage of the manuscript/Published Work that was contributed by the candidate:	99.99%
and	
• Describe the contribution that the candidate has made to the Manuscript/Published Work:	
Completed solely except utilizing feedback provided by the secondary author in the book Chapter: Robinson, D. & Scogings, C. (2017). Picking High Level Fruit in Dark Trees: Using Complex Systems Analytics to Detect and Understand Crime. In A. Colarik, J. Ang-Jaccard & A. Mathrani (Eds.), Cyber Security and Policy: A substantive dialogue (pp. 87-108). Auckland: Massey University Press.	
For manuscripts intended for publication please indicate target journal:	
Candidate's Signature:	
Date:	16/05/2019
Primary Supervisor's Signature:	
Date:	20/05/2019

(This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis)

MASSEY UNIVERSITY
GRADUATE RESEARCH SCHOOL

STATEMENT OF CONTRIBUTION
DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the candidate and the candidate's Primary Supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the candidate's contribution as indicated below in the *Statement of Originality*.

Name of candidate:	David Robinson
Name/title of Primary Supervisor:	Chris Scoglings
Name of Research Output and full reference:	
Robinson, D., & Scoglings, C. (2018). The detection of criminal groups in real-world fused data using the graph-mining algorithm 'GraphExtract'. Security Informatics, 7(2), 1.	
In which Chapter is the Manuscript / Published work:	NA
Please indicate:	
• The percentage of the manuscript/Published Work that was contributed by the candidate:	99.99%
and	
• Describe the contribution that the candidate has made to the Manuscript/Published Work:	
Completed every aspect utilizing feedback provided by the secondary author. In the case: Robinson, D., & Scoglings, C. (2018). The detection of criminal groups in real-world fused data using the graph-mining algorithm 'GraphExtract'. Security Informatics, 7(2), 1.	
For manuscripts intended for publication please indicate target journal:	
Candidate's Signature:	
Date:	16/05/2019
Primary Supervisor's Signature:	
Date:	20/05/2019

(This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an Appendix at the end of the thesis)