SIIMCO: A Forensic Investigation Tool for Identifying the Influential Members of a Criminal Organization

Kamal Taha, *Senior Member, IEEE*, and Paul D. Yoo, *Senior Member, IEEE*

Abstract— **Members of a criminal organization, who hold central positions in the organization, are usually targeted by criminal investigators for removal or surveillance. This is because they play key and influential roles by acting as commanders who issue instructions or serve as gatekeepers. Removing these central members (i.e., influential members) is most likely to disrupt the organization and put it out of business. Most often, criminal investigators are even more interested in knowing the portion of these influential members, who are the immediate leaders of lower-level criminals. These lower-level criminals are the ones who usually carry out the criminal works; therefore, they are easier to identify. The ultimate goal of investigators is to identify the immediate leaders of these lower-level criminals in order to disrupt future crimes. We propose in this paper a forensic analysis system called SIIMCO that can identify the influential members of a criminal organization. Given a list of lower-level criminals in a criminal organization, SIIMCO can also identify the immediate leaders of these criminals. SIIMCO first constructs a network representing a criminal organization from either Mobile Communication Data that belongs to the organization or from crime incident reports. It adopts the concept space approach to automatically construct a network from crime incident reports. In such a network, a vertex represents an individual criminal and a link represents the relationship between two criminals. SIIMCO employs formulas that quantify the degree of influence/importance of each vertex in the network relative to all other vertices. We present these formulas through a series of refinements. All the formulas incorporate novel-weighting schemes for the edges of networks. We evaluated the quality of SIIMCO by comparing it experimentally with two other systems. Results showed marked improvement.**

***Index Terms*— Forensic investigation, digital forensic, forensic analysis, criminal network, social network, mobile communication data, relative importance, central nodes.**

I. Introduction

Social network analysis (SNA) has long been used for identifying social groups and for determining the relationships among the members of social groups [3, 47]. In SNA, a network that depicts a social group needs to be constructed first and then analyzed. In such a network, a vertex represents a member of a social group and an edge represents the relationship between two members (e.g., it relates two vertices). Mobile Communication Data (MCD) is a way of constructing a social network, where a vertex in the network represents a person (i.e., a contact) and an edge represents a flow of communications between two persons (e.g., phone call records, messages, etc.). The communication records are collected either directly from mobile devices or indirectly from mobile network providers. Researchers from multidisciplinary fields took advantage of the valuable information contained in MCD to infer useful patterns and trends. For example, Urban Planning engineers analyze MCD to understand driving behavior and determine areas of congestion [17]. Health Care researchers analyze MCD to understand the correlation between human mobility and the spreading of infectious diseases in a specific geographical area [1].

Usually, forensic investigators aim at identifying individuals who are involved or can potentially be involved in a criminal activity [22]. Digital Forensics has emerged as a promising tool for forensic investigators. Usually, forensic investigators analyze communication records to infer the relationships among criminal suspects. In recent years, forensic investigators have shown a significantly growing interest on employing MCD for detecting criminal communities and identifying the influential members of these communities [22]. Usually, criminals involved in organized crimes, such as drug trafficking, terrorism, and criminal gangs, plot their activities through mobile phone communications [22]. For example, drug traffickers most often communicate with each other through mobile phones to contemplate and arrange for the smuggling, distributing, and selling of drugs [22]. The relationships among these criminals can be modeled as a network, where each vertex in the network represents a criminal and his specific role in the crime and each edge represents the communication attempts between two criminals. Therefore, it is imperative for forensic investigators to analyze such a network to determine how criminals are related and to identify the influential members of a criminal organization. Towards this, we analyze the structure of the network (i.e., how the communications between the criminals flow). In Digital Forensics, MCD has also been used extensively for identifying the dynamics of criminal networks.

After constructing a network that depicts the members of a criminal organization and their relationships, forensic investigators usually attempt to identify the *relative importance* of each vertex in the network to identify the most important vertices representing the influential members of the organization. In literature, this is known as the *relative importance* problem [21]. A large number of methods have been developed in recent years for determining the relative importance of vertices. Most of these methods employ standard network metrics techniques, *k*-clique techniques [20, 23, 30], or semantic similarities techniques [8].

Most often, criminal investigators are even more interested in knowing a portion of the influential members of a criminal organization, who are the immediate leaders of lower-level criminals. These lower-level criminals are the ones who usually carry out the criminal works; therefore, they are easier to identify (e.g., easier to implicate and arrest). The ultimate goal of investigators is to identify the immediate leaders of these lower-level criminals in order to disrupt future crimes. Given a list of

K. Taha is with the Electrical and Computer Engineering Department, Khalifa University, UAE (e-mail: kamal.taha@kustar.ac.ae).

P. Yoo is with the Faculty of Science and Technology, Bournemouth University, United Kingdom (email: pyoo@bournemouth.ac.uk).

vertices representing lower-level criminals in a network depicting a criminal organization, forensic criminal investigators would try to identify the most important vertices to this list of vertices. These vertices would represent the immediate leaders to the given list of lower-level criminals. The vertices in a network that represents lower-level criminals are known as *query vertices*.

We propose in this paper a forensic analysis system called **SIIMCO** (**S**ystem for **I**dentifying the **I**nfluential **M**embers of a **C**riminal **O**rganization). SIIMCO can identify the most influential members of a criminal organization. Given a list of lower-level criminals in a criminal organization, SIIMCO can also identify the immediate leaders of these lower-level criminals. Identifying the influential members of a criminal organization is one of the most important tasks that criminal investigators undertake. Usually, members of a criminal organization, who hold central positions in a criminal organization, are targeted by criminal investigators for removal or surveillance [5, 31]. This is because these central members (i.e., influential members) usually play key and influential roles in the organization by either acting as commanders who issue instructions to other members or serving as gatekeepers, who receive and distribute information and goods to other members. Removing these central members is most likely to disrupt the organization and put it out of business. Shang et al. [39] stated that a common problem in a criminal investigation that involves a criminal organization is to identify the leaders of the organization before they make an arrest. Memon [33] stated that the identification of key actor(s) in criminal covert networks is a major objective for criminal investigators. Memon also stated that by isolating or eliminating key actors in a criminal network, the network can be destabilized or, at least, its ability to effectively function can be significantly reduced. Wiil et al. [46] stated that the identification and elimination of key nodes in a terrorist network will decrease the ability of the network to function normally. Investigators always know several members of a conspiracy, but hope to identify the leaders before they make arrests [2, 48]. The way to put an end to a criminal organization traditionally was to arrest the leaders, thereby incapacitating the remaining bad guys that did most of the leg work [27].

Despite the success of most current methods for identifying the vertices that are important to query vertices, these methods suffer incomplete contribution and inconsistent contribution. Incomplete contribution occurs, if some query vertices do not contribute to the overall relative importance value of a vertex. The inconsistent contribution occurs, if query vertices contribute unequally to the overall relative importance value of a vertex. Let *v* be the current vertex under consideration. SIIMCO overcomes the problem of Incomplete Contribution by: (1) considering the importance of *each* query vertex to *v*, and (2) assigning a weight to each incoming edge to *v* that is outgoing from one of the query vertices (this weight represents the importance/rank of this vertex relative to all incoming edges to *v*). SIIMCO overcomes the problem of Inconsistent Contribution by: (1) considering the importance of *each* query vertex to each vertex connected to *v*, and (2) accounting for the degree of relativity of *v* to all query vertices. That is, SIIMCO overcomes the incomplete and inconsistent contribution limitations of most current methods outlined above.

In the framework of SIIMCO, a network can be constructed from MCD that belongs to a criminal organization. In such a network, a vertex represents a criminal (i.e., a caller/receiver) and an edge represents a flow of communications/information between two criminals (e.g., phone call records, messages, etc.). A network can also be constructed from crime incident reports. These reports usually include the names of criminals/suspects, the type of crime, and the location and date of the crime. We assume that criminals who appear in the same crime incident report collaborate in committing crimes. We also assume that the more criminals appear in the same crime incident reports the stronger their relationships are. Thus, the number of co-occurrences of criminals' names in the same crime incident reports can be considered indicative of the strength/weight of the relationships between these criminals [19]. SIIMCO adopts the concept space approach [9] to construct a network automatically from crime incident reports [10]. In such networks, a vertex represents an individual criminal, a link represents the relationship between two criminals, and a co-occurrence weight of a link represents a relational strength between two criminals.

SIIMCO identifies the influential members of a criminal organization using a formula that quantifies the degree of influence/importance of each criminal in a criminal organization relative to all other criminals in the organization. The formula identifies the central (i.e., influential) members by determining the vertices that represent them in the network depicting the criminal organization. In this paper, we present this formula through a series of enhancements and improvement refinements. SIIMCO also identifies the immediate leaders of a given list of lower-level criminals using a formula that quantifies the degree of influence/importance of each criminal in the criminal organization on the given list of lower-level criminals (i.e., query vertices). Given a set of query vertices representing lower-level criminals, SIIMCO determines the relative importance of each vertex in the network with respect to the query vertices. In this paper, we also present the formula that identifies the immediate leaders of lower-level criminals through a series of enhancements and improvement refinements. All the formulas incorporate novel weighting schemes for the edges of networks. Let v be a vertex under consideration. The formula takes into consideration several factors such as the importance of v, the importance of the vertices connected with v, the degree of relativity of v to all query vertices, and the ranking of all vertices based on their overall importance in the network.

II. Related Work

A number of methods have been proposed to identify the set of suspicious source nodes (e.g., fake followers, botnets, etc.) on a given criminal network. For example, Meng et al. [32] introduced the concepts of "synchronized" and "abnormal" nodes to investigate the behavior patterns of source nodes in a criminal network in order to identify suspicious nodes. The authors spot suspicious nodes by plotting synchronicity-normality points. Source nodes are considered synchronized if they have very similar behavior patterns. Source nodes are considered abnormal if their behavior pattern is very different from the majority of other nodes. The authors define the synchronicity of a node *u* as the average closeness between each pair of *u*'s targets, as shown in Equation 1. The authors define the normality of a node *u* as the average closeness between each pair of *u*'s targets and other nodes, as shown in Equation 2.

$$sync(u) = \frac{\sum_{(v,v')\in O(u)\times O(u)} c(v,v')}{d_o(u)\times d_o(u)} \quad (1)$$

$$norm(u) = \frac{\sum_{(v,v') \in O(u) \times u} c(v,v')}{d_o(u) \times N} \quad (2)$$

- $c(v,v')$: The closeness (similarity) between two target nodes v and v'.
- $O(u)$: The set of u's target nodes.
- $d_o(u)$: The out-degree of node u, which is the number of its targets, i.e. the size of $O(u)$
- u: The set of nodes.
- N: The number of nodes in set u.

Recently, structural analysis techniques have been used for investigating and identifying criminal communities [31]. For example, CrimeNet Explorer [22] divides a network depicting a criminal organization into subgroups based on their strength of relations, using hierarchical clustering technique. To identify the influential members of a subgroup, it employs the well-known Degree Centrality, Closeness Centrality, and Betweenness Centrality metrics. To determine the degree of relationship between two vertices, it uses the shortest path algorithm and Blockmodeling [4]. In 2013, Catanese et al. [11] introduced an initial version of a system called LogAnalysis. The system was intended for forensic visual statistical analysis of mobile phone logs. In this initial version of LogAnalysis, the relationships among mobile phone users are represented graphically, which helps in understanding the hierarchies within criminal organizations and discovering key and central members inside those organizations [11]. This also helps to visually discover gangs, by measuring their cohesion in terms of the density of internal connections [11]. This is done by exploiting the centrality measures provided by the Social Network Analysis. The system can also help in analyzing temporal information from phone call networks [11]. In 2014, LogAnalysis was improved and enhanced to identify the influential criminals in a criminal organization by determining the degree of relationships between the vertices representing criminals [14]. The enhanced version of LogAnalysis uses the Girvan and Newman [18] and Newman [34] algorithms. It uses the Girvan & Newman algorithm to compute edge betweenness. It uses the Newman algorithm to cluster the network hierarchically using a greedy strategy by aggregating vertices to form tighter sub-communities. A sub-community is determined by considering the entire topology of the network. The system applies statistical metrics to analyze the relationships among vertices to identify the sub-communities that represent tied relationships among their members.

Kleinberg [26] proposed a technique for locating high-quality information using a structural analysis of the link topology surrounding authoritative nodes in a graph. The techniques can be applied for identifying authoritative nodes in any graph, even though the authors believe that the technique is more compelling in the context of the Web for discovering the most authoritative webpages in a specific search topic. The authors observe that a certain natural type of equilibrium exists between hub nodes and authority nodes in a graph defined by the link structure, and they construct an algorithm for identifying important nodes using this observation. A hub is a node that links many related authority nodes. An authority node is a node that is pointed to by many hubs.

Other network analysis techniques have been used for detecting communities. One of these techniques is called *k*-clique. The term "clique" was introduced by Luce and Perry [23]. A clique is a graph/network, whose vertices are connected. Each *k*-clique subnetwork represents a community [29]. The input to a system employing the *k*-clique technique is a network along with a value *k*, and the output is a clique of size *k*. The authors in [20] used the k-clique technique for identifying the communities involved in the so-called Nigerian fraud scamming. The authors linked the email addresses of the scammers to Facebook profiles. The result was a social network consisting of over 40,000 vertices. Then, the authors transformed the network into 7-clique and 6-clique communities. The authors demonstrated strong ties among vertices in each clique. Similarly, the authors of [30] employed the *k*-clique technique to investigate the relationships among a community of hackers called Shadowcrew.

Social network analysis (SNA) has long been used for identifying social groups and for determining the relationships among the members of social groups [3, 47]. In recent years, digital forensic investigators have shown a significantly growing interest in employing similar network analysis techniques for detecting criminal communities and identifying the influential members of these communities [22]. Relational analysis is mainly used for determining the central/important vertices in a social network (i.e., the influential members of a social group). This is done by assigning weights to edges to reflect the relational strength of the vertices connected by the edges. The metrics used in Relational Analysis can be classified as Degree Centrality, Closeness Centrality, or Betweenness Centrality. Degree is the number of ties that a vertex has. Vertices with a high degree of centrality are central in the network. The degree centrality is calculated as shown in Equation 3, where *n* is the number of vertices in the network and x_{uv} equals 1 if vertices *v* and *u* are connected and 0 otherwise.

$$C_I(u) = \sum_{v \neq u}^{n} x_{vu} \quad (3)$$

The betweenness centrality of a vertex is computed based on the number of shortest paths between other vertices that pass through this vertex. The betweenness centrality is calculated as shown in Equation 4, where σ_{st} is the number of shortest paths between vertex *s* and vertex *t*, and $\sigma_{st}(u)$ is the number of shortest paths between vertex *s* and vertex *t* that pass through *u*.

$$C_B(u) = \sum_{s,t \in G, s \neq t} \frac{\sigma_{st}(u)}{\sigma_{st}} \quad (4)$$

Closeness is the length of the shortest path to all other vertices. It measures how a vertex is close to other vertices. The closeness centrality is calculated as shown in Equation 5, where $d(u_i, u_j)$ is the distance between vertex u_i and vertex u_j.

$$C_C(u_i) = \frac{1}{\sum_{j \neq i}^{n} d(u_i, u_j)} \quad (5)$$

Other Relational Analysis techniques have been used in different fields. For example, they have been used for determining the semantic similarities of the members of a social group [43, 44, 45]. Social Network Positional Analysis considers the overall structure of a social network. The key method for Positional Analysis is blockmodeling [4]. Blockmodeling involves the following two steps: (1) network partitioning, and (2) interaction pattern identification. In network partitioning, a network is divided into positions based on the structural similarities among the vertices of the network [28]. Similar techniques have also been used in the area of bioinformatics [40, 42].

III. Network Creation

The structure of a network can be a valuable intelligence tool that can help digital forensic investigators to infer important information to identify criminals and their leaders and to disrupt future criminal acts. Usually, network topologies can be classified based on their structures into hierarchical structure, cellular structure, flat structure, chain structure, and star structure.

In hierarchical structure (e.g., [37]), vertices are clustered based on their degree of relationships. In cellular structure [41], the network consists of strongly related subgroups connected by bridges. Vertices and edges have the same cellular component types. All vertices are strongly connected with one another so that information between any two vertices can flow directly. In flat structure [25], each vertex is connected with other vertices either directly or through a few vertices. In chain structures [15], vertices are connected one by one so that information can flow from one vertex to another. In star structures [5], vertices are all connected to a central vertex, which acts as a hub. That is, the network has a centralized structure. The removal of the central vertex can disrupt the network. In the framework of SIIMCO, a hybrid hierarchical-flat structure is adopted as a network topology. In subsections *A* and *B*, we describe two approaches employed by SIIMCO for gathering the information needed to construct networks.

A. Creating a Network from Mobile Communication Data belongs to Criminals

In the framework of SIIMCO, a network can be constructed from MCD that belongs to a criminal organization. In such a network, a vertex represents a criminal (i.e., a caller/receiver) and an edge represents a flow of communications/information between two criminals (e.g., phone call records, messages, etc.). That is, an edge relates two vertices in the network to each other. The weight of an edge connecting two vertices represents the number of calls/messages between the two criminals represented by the two vertices. Thus, a weight of an edge reflects the relational strength of the vertices connected by the edge. The communication records are collected either directly from mobile devices that belong to the criminals or indirectly from mobile network providers. We denote a network constructed from MCD as $N = (V, E)$, where V is a (finite) set of vertices representing criminals, and E is a (finite) set of edges connecting vertices.

B. Creating a Network from Crime Incident Reports

In the framework of SIIMCO, a network can also be constructed from crime incident reports. These reports usually include the names of criminals/suspects, the type of crime, and the location and date of the crime. We assume that criminals who appear in the same crime incident report collaborate in committing crimes. We also assume that the more criminals appear in the same crime incident reports the stronger their relationships are. Thus, the number of co-occurrences of criminals' names in the same crime incident reports can be considered as indicative of the strength/weight of the relationships between these criminals [19]. This is because the co-occurrence of criminals' names in the same reports can reveal certain patterns, which can be transformed to relationships (i.e., edges) in the network. In such networks, a vertex represents an individual criminal, a link represents the relationship between two criminals, and a co-occurrence weight of a link represents a relational strength between two criminals. The weights of links are normalized to a value in the range between 0 and 1.

Constructing networks by manually extracting relational data from crime incident reports can be very time-consuming. Therefore, SIIMCO adopts the concept space approach [10] to construct networks automatically from crime incident reports [13]. SIIMCO employs this concept to identify the relationships among criminals and transforms these relationships into networks. This concept identifies related words/phrases based on their co-occurrences in the same documents (e.g., crime incidents). The names of each two criminals/suspects in a crime incident report represents a pair. A pair of words/phrases is determined to be related based on the frequency of their co-occurrences in the sentences of the same crime incidents. The relationship of each pair is assigned a weight that reflects the strength of the relationship. This weight is determined based on the statistical significance of the co-occurrences of the pair in crime incident reports. That is, the more the pair co-occur the more related it is.

The concept space approach has the advantage of preventing extremely large co-occurrence weights from being skewed [9]. SIIMCO can accept as an input any criminal-related document such as a crime incident report, financial transaction records, and telephone records. SIIMCO uses Stanford Named Entity Recognition (NER) [38] techniques for identifying the names of criminals/suspects in reports. SIIMCO uses a tokenizer and stemmer to align a sequence of words and the names of persons. A person's stemmed words are aligned against input reports.

C. Constructing an Algorithm for Creating a Network

We constructed an algorithm called CONST-NW (see Fig. 1) that constructs a network from an input MCD or a crime incident report. The algorithm constructs a network in terms of adjacency lists. That is, it represents a network in terms of the adjacency lists of its vertices. An adjacency list is a linked list that identifies all the vertices to which a particular vertex is connected. Algorithm CONST-NW stores an input network's set of vertices V in a queue called Q. The adjacencies of each vertex $u \in V$ are stored in an array called $Adj[u]$ (see lines 3 and 8). The parents of each vertex $v \in V$ are stored in an array called $\pi[v]$. The frequency of co-occurrences of the pair u and v in criminal incident reports are normalized by dividing it by the sum of the overall frequencies n and stored in a variable called $w(u, v)$ as the weight of the edge (u, v) (see line 10). The adjacency lists that represents the network are stored in a queue called *NW* (see line 11).

The network information are then stored in database tables. Each table stores the information of a vertex $u \in V$. Each tuple holds each vertex $v \in Adj[u]$, the weight $w(u, v)$, and the parents of u. The stored data will be used later for structural analysis and for network visualization.

CONST-NW

```
1.  for each vertex u ∈ V
2.          π[u] ← NIL
3.          Adj[u] ← NIL
4.  Q ← Ø
5.  ENQUEUE(Q)
6.  while Q ≠ Ø
7.      do u ← DEQUEUE(Q)
8.           Adj[u] ← v
9.           π[v] ← u
10.          w(u, v) ← Occurrences(u, v)/n
11.          ENQUEUE(NW, Adj[u])
```

Fig. 1: Algorithm *CONST-NW*

IV. Identifying the Influential Members of Criminal Networks and Criminal Sub-Networks

Members of a criminal organization, who hold central positions in the organization, are targeted by criminal investigators for removal or surveillance [5, 31]. This is because these central members usually play key and important roles in the organization by acting as commanders who issue instructions to other members or serving as gatekeepers, who receive and distribute information and goods to other members. Removing these central members is most likely way to disrupt the organization and put it out of business. In subsection *A*, we construct a formula that identifies these influential/important members by determining the vertices representing them in a network depicting the criminal organization.

Often, investigators are even more interested in knowing a portion of the influential members of a criminal organization, who are the immediate leaders of lower-level criminals. These lower-level criminals are the ones who usually carry out the criminal works and are therefore easier to identify (e.g., easier to implicate and arrest). The ultimate goal of investigators is to identify the leaders of these lower-level criminals in order to disrupt future crimes. In subsection *B*, we construct a formula that identifies the immediate leaders to a given list of lower-level criminals under investigation. The formula identifies the most important vertices to a given list of vertices representing lower-level criminals in a network depicting a criminal organization.

A. Identifying the Influential Members of a Criminal Organization

We construct a formula that quantifies the degree of influence/importance of each criminal in a criminal organization relative to all other criminals in the organization. We construct the formula through two refinements. We introduce the initial formula in subsection 1. We refine and optimize it in subsection 2.

1) Assigning Weight to a Vertex based on its Number of Communication Attempts

In this subsection, we construct an initial formula that assigns a weight to each vertex v_k to reflect its importance in the network relative to all other vertices. The weight of v_k is determined based on the following:

a) The number of incoming and outgoing phone calls/messages to v_k (or, the number of occurrences of other vertices in crime incident reports associated with v_k, and the number of occurrences of vertex v_k in crime incident reports associated with other vertices).

b) The number of incoming and outgoing edges to v_k (i.e., the number of vertices that have outgoing edges to v_k, and the number of vertices that have incoming edges from vertex v_k).

We compute the weights as shown in Equation 6.

$$w(v_k) = \frac{\sum_{i=1}^{|v_k^E(in)|} |(v_i, v_k)| + \sum_{j=1}^{|v_k^E(out)|} |(v_k, v_j)|}{\sum_{i=1}^{|v_k^E(in)|} 0.8\,|(v_i, v_k)| + \sum_{j=1}^{|v_k^E(out)|} 0.6\,|(v_k, v_j)|} \quad (6)$$

- $w(v_k)$: Weight of the current vertex under consideration (i.e., vertex v_k).
- $|(v_i, v_k)|$: Number of incoming phone calls/messages to vertex v_k from a vertex v_i. Or, the number of occurrences of a vertex v_i in crime incident reports associated with v_k
- $|(v_k, v_j)|$: Number of outgoing phone calls/messages from vertex v_k to a vertex v_j. Or, the number of occurrences of vertex v_k in crime incident reports associated with a vertex v_j.
- $|v_k^E(in)|$: Number of incoming edges to vertex v_k (i.e., number of vertices that have outgoing edges to vertex v_k).
- $|v_k^E(out)|$: Number of outgoing edges from vertex v_k (i.e., number of vertices have incoming edges from v_k).

2) Improving Equation 6 by Assigning Weights to Edges that Represent their Significance to the Vertex under Consideration

In this subsection, we refine and optimize Equation 6 by considering the weights of edges as a measure of importance to vertices. Towards this, we represent each incoming edge and outgoing edge to/from a current vertex under consideration v_k by a weight that reflects its importance to v_k. We assign a weight to each incoming edge to v_k to represent its importance/rank to v_k relative to all incoming edges to v_k. Similarly, we assign a weight to each outgoing edge from v_k to represent its importance/rank to v_k relative to all outgoing edges from v_k. We revised Equation 6 accordingly as shown in Equation 7:

$$w(v_k) = \frac{\sum_{i=1}^{|v_k^E(in)|} w(v_i, v_k) + \sum_{j=1}^{|v_k^E(out)|} w(v_k, v_j)}{\sum_{i=1}^{|v_k^E(in)|} 0.8\ w(v_i, v_k) + \sum_{j=1}^{|v_k^E(out)|} 0.6\ w(v_k, v_j)} \quad (7)$$

where $w(v_i, v_k)$ and $w(v_k, v_j)$ are computed as shown in Equations 8 and 9 respectively.

$$w(v_i, v_k) = \log\left(1 + \frac{\max v_k(in)}{|(v_i, v_k)|}\right) \times \frac{1}{|v_k^E(in)|} \quad (8)$$

$$w(v_k, v_j) = \log\left(1 + \frac{\max v_k(out)}{|(v_k, v_j)|}\right) \times \frac{1}{|v_k^E(out)|} \quad (9)$$

- $w(v_i, v_k)$: Weight of an incoming edge to the current vertex under consideration v_k from a vertex v_i.
- $w(v_k, v_j)$: Weight of an outgoing edge from the current vertex under consideration v_k to a vertex v_j.
- $\max v_k(in)$: Maximum number of incoming phone calls/messages to vertex v_k from one of the vertices. Or, the maximum number of occurrences of one of the vertices in a crime incident report associated with v_k.
- $\max v_k(out)$: Maximum number of outgoing phone calls/messages from v_k to another vertex. Or, the maximum number of occurrences of v_k in a crime incident report associated with one of the other vertices.

Vertices are ranked based on their weights computed using Equation 7. Criminals represented by the top-*k* vertices are considered the influential members of the criminal organization.

B. Identifying the Immediate Leaders of Lower Level Criminals Under Investigation

Criminal investigators sometimes want to know the immediate leaders of lower-level criminals, who carry out crimes and are under investigation (e.g., under arrest). In this section, we refer to the vertices in the network that represent these lower-level criminals as *query vertices*. We construct a formula that quantifies the degree of influence of each criminal in a criminal organization on a given list of lower-level criminals under investigation (i.e., query vertices). We construct the formula through two refinements. We introduce the initial formula in subsection 1. We refine it in subsection 2. In subsection 3, we describe how the top immediate leaders of the given list of lower-level criminals are determined.

1) Adjusting Equation 7 by Considering the Weights of Vertices connected with Query Vertices

To identify the most important vertices to query vertices, we adjusted Equation 7 by considering the following: (1) the importance of each query vertex to the current vertex under consideration v_k, and (2) the importance of each query vertex to each vertex connected with v_k. Accordingly, we revised Equation 7 as shown in Equation 10.

$$w(v_k) = \frac{\sum_{i=1}^{|v_k^E(in)|} w(v_i, v_k) \times w(v_i) + \sum_{j=1}^{|v_k^E(out)|} w(v_k, v_j) \times w(v_j)}{\sum_{i=1}^{|v_k^E(in)|} 0.8\ w(v_i, v_k) \times w(v_i) + \sum_{j=1}^{|v_k^E(out)|} 0.6\ w(v_k, v_j) \times w(v_j)} \quad (10)$$

where $w(v_i)$ and $w(v_j)$ are computed as shown in Equations 11 and 12.

$$w(v_i) = \sum_{z=1}^{|Q|} \left((1 - p_{out}^{q_z} - p_{in}^{q_z})\, w(q_z) + p_{in}^{q_z} \sum \frac{S_{q_z}(v_i)}{|v_i^C(in)|} + p_{out}^{q_z} \sum \frac{S_{q_z}(v_i)}{|v_i^C(out)|} \right) \quad (11)$$

$$w(v_j) = \sum_{z=1}^{|Q|} \left((1 - p_{out}^{q_z} - p_{in}^{q_z})\, w(q_z) + p_{in}^{q_z} \sum \frac{S_{q_z}(v_j)}{|v_j^C(in)|} + p_{out}^{q_z} \sum \frac{S_{q_z}(v_j)}{|v_j^C(out)|} \right) \quad (12)$$

- $w(v_i)$: Weight of vertex v_i, which has an incoming edge to vertex v_k.
- $w(v_j)$: Weight of vertex v_j, which has an outgoing edge from vertex v_k.
- $|Q|$: Number of query vertices.
- $w(q_z)$: Weight of query vertex $q_z \in Q$, calculated using Equation 7.
- $p_{in}^{q_z}$: A discretionary parameter denotes the importance of incoming edges to vertex v_i originated from q_z. The value of $p_{in}^{q_z}$ is determined heuristically. In our experiments, we set the value of $p_{in}^{q_z}$ to 0.5.
- $p_{out}^{q_z}$: A discretionary parameter denotes the importance of outgoing edges from v_i to q_z. The value of $p_{out}^{q_z}$ is determined heuristically. In our experiments, we set the value of $p_{out}^{q_z}$ to 0.3.
- $|v_i^C(in)|$: Number of incoming phone calls/messages to vertex v_i. Or, the number of occurrences of other criminals in crime incident reports associated with v_i.

- $|v_i^C(out)|$: Number of outgoing phone calls/messages from vertex v_i. Or, the number of occurrences of v_i in crime incident reports associated with other criminals.
- $S_{q_z}(v_i)$: The score of vertex v_i with regards to query vertex q_z. It is computed as follows. Let: (1) "*a*" be the number of incoming and outgoing phone calls/messages to and from query vertex q_z that involve vertex v_i (or, the number of occurrences of q_z in crime incident reports associated with v_i), (2) "*b*" be the number of incoming and outgoing phone calls/messages to and from vertex v_i that involve q_z (or, the number of occurrences of v_i in reports associated with q_z), and (3) "*c*" be the number of incoming and outgoing phone calls/messages to and from q_z. The score $S_{q_z}(v_i)$ is the probability that the number of incoming and outgoing phone calls/messages to and from q_z that involve vertex v_i (or, the number of occurrences of q_z in crime incident reports associated with v_i) is exactly "*k*" out of the "*c*", and it is given by the following Fisher's exact test [7]:

$$S(v_i) = \sum_{i=0}^{k-1} \frac{\binom{b}{i}\binom{a-b}{c-i}}{\binom{a}{c}} \qquad (13)$$

2) Improving Equation 10 by Accounting for the Degree of Relativity of the Vertex under Consideration to Query Vertices

We improve Equation 10 by accounting for the degree of relativity of the current vertex under consideration to all the query vertices. Towards this, we first compute the summation of the distances *d* from the current vertex under consideration to all query vertices. Then, we *penalize and scale down* the weight of the current vertex under consideration relative to its distance from the query vertices, by a factor of $decay^{d-1}$, where *decay* is a parameter that can be set to a value in the range 0 to 1. Therefore, we adjusted Equation 10 as shown in Equation 14:

$$w(v_k) = \frac{\sum_{i=1}^{|v_k^E(in)|} w(v_i, v_k) \times w(v_i) + \sum_{j=1}^{|v_k^E(out)|} w(v_k, v_j) \times w(v_j)}{\sum_{i=1}^{|v_k^E(in)|} 0.8\ w(v_i, v_k) \times w(v_i) + \sum_{j=1}^{|v_k^E(out)|} 0.6\ w(v_k, v_j) \times w(v_j)} \times decay^{d-1} \qquad (14)$$

- *Decay*: A parameter that can be set to a value in the range 0 to 1.
- d: The summation of the distances from the current vertex under consideration to all query vertices. It is an exponent that accounts for the degree of relativity of the current vertex under consideration to the query vertices.

3) Identifying the Top Immediate Leaders

Vertices are ranked based on their weights computed in Equation 14. A criminal represented by vertex v_k is considered an immediate leader to the lower-level criminals represented by the query vertices, if he or she acquires a weight $w(v_k)$ greater than a threshold β, which is the value lower than the mean weight by the standard error of the normalized mean. β is computed as follows:

$$\beta = \frac{1 - \sqrt{\sum_{\forall v \in V} (w(v_k) - \frac{1}{|V|})^2}}{|V|} \qquad (15)$$

- $|V|$: Set of all vertices.

C. System Architecture

Fig. 2 shows the system architecture. The module *Network Creator* receives either MCD or crime incident reports and outputs a corresponding network. The module *Network Creator* includes Stanford NER for identifying the names of criminals/suspects in crime incident reports. The module uses algorithm *CONST-NW* shown in Fig. 1 to construct a network in terms of the adjacency lists of its vertices. The module *Influential Member Locator* receives a network and determines the most influential node in the network using Equation 7, which quantifies the degree of influence/importance of each node relative to all other nodes in the network. The module *Immediate Leader Locator* receives a network and determines the immediate leader of lower-level criminals using Equation 14, which quantifies the degree of influence of each node to a given list of query nodes.

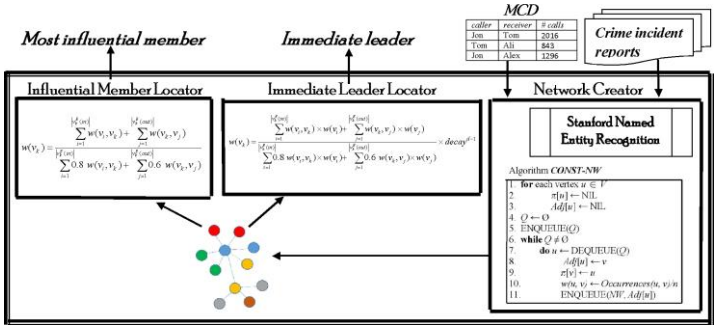


Fig. 2: System architecture

V. Experimental Results

We implemented SIIMCO in Java, ran it under Windows 8 and an Intel(R) Core(TM) i5-4200U processor, using a CPU of 2.30 GHz and RAM of 4 GB. In this section, we evaluate the quality of SIIMCO by comparing it experimentally with CrimeNet Explorer [22] and LogAnalysis [14]. We evaluate the three systems using two real-world communication records. We also used DBLP dataset [12]. We aim at evaluating and comparing the accuracy of the three systems for identifying the following:

1. The influential/important members of a criminal organization. Towards this, we evaluate the accuracy of the

three systems for identifying the most important vertices in the two real-word networks.

2. The immediate leaders of a given list of lower-level criminals in a criminal organization. Towards this, we evaluate the accuracy of the three systems for identifying the most important vertices to a given list of vertices (i.e., query vertices) in the real-word networks.

The following are brief overviews of CrimeNet Explorer [22] and LogAnalysis [14]:

- ***CrimeNet Explorer [22]:*** Given a network, whose vertices represent criminals, CrimeNet Explorer divides the network into subgroups based on their strength of relations, using hierarchical clustering technique. To identify the influential (i.e., important) members of a subgroup, CrimeNet Explorer employs the well-known Degree, Closeness, and Betweenness Centrality metrics. To determine the degree of relationship between two vertices, it uses the shortest path algorithm and Blockmodeling [4].

- ***LogAnalysis [14]:*** Given a network, whose vertices represent criminals and edges represent mobile phone communications between the criminals, LogAnalysis identifies the influential (i.e., important) criminals by determining the degree of relationships between the vertices. Towards this, it uses the Girvan and Newman [18] and Newman [34] algorithms. It uses the Girvan and Newman algorithm to compute edge betweenness. It uses the Newman algorithm to cluster the network hierarchically using a greedy strategy by aggregating vertices to form tighter sub-communities. A partition (i.e., a sub-community) is determined by considering the entire topology of the network.

A. *Compiling Datasets for the Evaluation*

For the evaluation we use two real-world communication datasets: Enron email corpus [13, 16, 24] and Nodobo mobile phone records dataset [6, 34]. We also used DBLP [12]. We converted the two datasets into networks that depict the flow of information between users. Below are brief descriptions of the two datasets:

- ***Enron email dataset [13, 16, 24]:*** Enron email corpus are real-world internal email messages exchanged among Enron's employees and associates [13, 24]. Enron was the 7th largest business organization in USA. The email corpus surfaced after a criminal scandal involving top Enron employees was publicized. The crime started in 1999, when top employees and associates started to separate losses from equity and to derivate trades into "special purpose entities". Most of these emails revolve around this. The corpus contains an actual 619,446 email messages that belong to 158 Enron employees and associates, including senior Enron employees and associates. The dates of the emails are between 1998 to 2002. We cleaned the corpus by removing emails that were sent to or received from people other than the 158 employees. We also removed emails that are duplicate, junk, blank, or undelivered. The resulting dataset contains 200,136 emails from 151 Enron employees. We converted the email dataset into a network, where a vertex in the network represents one of the 151 employees. An edge represents email correspondences between two employees. The weight of an edge represents the number of emails exchanged between the two employees. The raw corpus is currently available online at [16].
- ***Nodobo mobile phone records dataset [6, 34]:*** Nodobo contains mobile phone records of 27 high-school students from September 2010 to February 2011. The dataset was compiled originally for studying mobile phone usage. The dataset includes 13,035 call records, 83,542 message records and 5,292,103 presence records. We constructed a network from the call and message records. In the network, a vertex represents one of the 27 students, an edge represents phone calls/messages between two students, and the weight of an edge represents the number of phone calls/messages between two students. The dataset is available online at [35].
- ***DBLP [12]:*** DBLP dataset is a network of co-authorship relationship between authors in the computer science field. In the experiment we used a partial snapshot of the original DBLP Bibliography where only the authors from 20 different conferences are considered [36]. We constructed a graph from the dataset where nodes represent authors and edges represent the number of papers in common between two authors. Since the relationship is symmetric, the graph is undirected.

B. *Evaluating the Accuracy of the Three Systems to Identify the Top Influential Vertices of a Whole Network*

1) Comparing the Systems' Results with Ground Truth Data obtained from Enron's Publicly Known Information

The investigation of Enron wrongdoing incriminated 28 Enron employees and associates. The names and identities of these 28 employees have been released to the public [13, 24]. The intensities of the emails sent and received by these 28 employees, intuitively, are proportionally the highest among the 151 employees in the Enron email dataset described in subsection V-A. This is because these 28 employees were in the center of the scandal and their emails revolved around contemplating and planning the crime. Thus, the vertices representing these 28 employees are the most central/important (i.e., influential) ones in the network depicting the Enron email dataset, due to the intensity of the emails sent and received by these employees. In this test, we consider the vertices representing these 28 employees to be ground truth data. We evaluate the accuracy of the three systems by comparing the influential/important vertices in the network returned by the three systems with the vertices representing the 28 employees. Based on this, we calculated the accuracy of each system in terms of Recall, Precision, and F-value, which are calculated as follows:

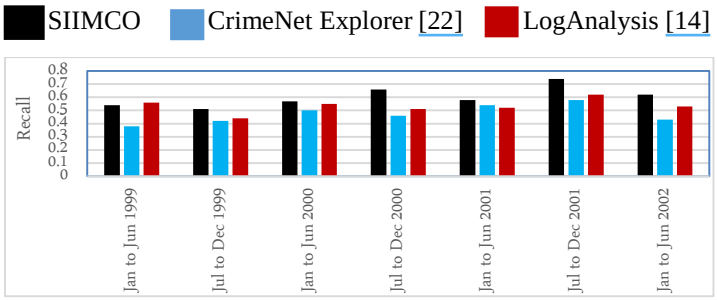
$$\text{Recall} = \frac{N_s^c}{N_m^{top}} \tag{16}$$

$$\text{Precision} = \frac{N_s^c}{N_s^{top}} \tag{17}$$

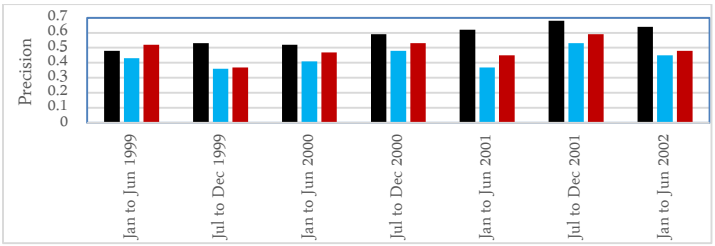
$$F-value = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{18}$$

- N_s^c : The number of *correct* vertices returned by a system (i.e., the number of vertices in the list returned by a system that are among the vertices representing the 28 employees).
- N_m^{top}: Equals 28 (the number of incriminated employees).
- N_s^{top}: The number of vertices in the list of influential/important vertices returned by a system.

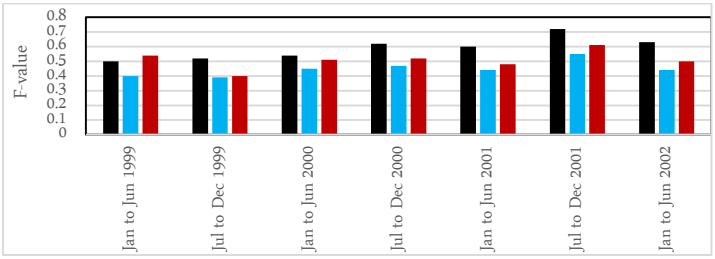
Figs. 3 show the Recall, Precision, and F-value of each system in each 6-month period from January 1999 to June 2002. Fig. 4 shows the *overall average* Recall, Precision, and F-value.



(a)



(b)



(c)

Figs. 3: (a) Recall, (b) Precision, and (c) F-value of the 3 systems on Enron dataset in each 6-month period from January 1999 to June 2002 using ground truth data obtained from Enron's publicly known information

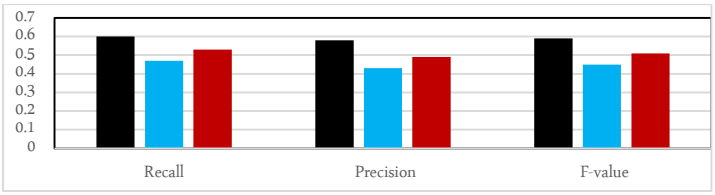


Fig. 4: The overall average Recall, Precision, and F-value of the 3 systems on Enron dataset using ground truth data obtained from Enron's publicly known information

2) Calculating the Recall, Precision, and F-value of the Systems by comparing their Results with Results Determined by Standard Network Metrics

In this test, we measure the performance of the systems by comparing their results with the results determined by the standard Closeness, Betweenness, In Degree, and Out Degree Centrality metrics. Recall section II for how the Closeness, Betweenness, and Degree Centralities are computed. For the sake of fair evaluation and comparison, we consider the same threshold β shown in Equation 15 for the results returned by the three systems and the four standard network metrics. We calculated the Recall, Precision, and F-value as shown in Equations 16-18. Let l_{top} be the list of important vertices determined by a standard network metric, whose weights are bigger than β. In Equations 16-18, N_s^c is the number of correct vertices returned by a system that are found in the list l_{top}, and N_m^{top} is the number of vertices in the list l_{top}.

We submitted the network representing Enron, Nodobo, and DBLP [12] datasets to the four standard network metrics, and we also submitted the same network to each of the three systems. We then calculated the Recall, Precision, and F-value of the results returned by each system. The results are shown in Tables 1-3.

TABLE I
PERFORMANCE OF THE THREE SYSTEMS ON THE ENRON DATASET BASED ON THE TOP VERTICES RETURNED BY THE STANDARD NETWORK METRICS

		Recall	Precision	F-value
SIIMCO	Closeness Centrality	0.62	0.56	0.59
CrimeNet Explorer		0.36	0.41	0.38
LogAnalysis		0.53	0.51	0.52
SIIMCO	Betweennes Centrality	0.56	0.52	0.54
CrimeNet Explorer		0.42	0.44	0.43
LogAnalysis		0.47	0.43	0.45
SIIMCO	In Degree Centrality	0.83	0.75	0.79
CrimeNet Explorer		0.55	0.49	0.52
LogAnalysis		0.66	0.64	0.65
SIIMCO	Out Degree Centrality	0.79	0.76	0.77
CrimeNet Explorer		0.53	0.55	0.54
LogAnalysis		0.64	0.58	0.61

TABLE 2
PERFORMANCE OF THE THREE SYSTEMS ON THE NODOBO DATASET BASED ON THE TOP VERTICES RETURNED BY THE STANDARD NETWORK METRICS

		Recall	Precision	F-value
SIIMCO	Closeness Centrality	0.67	0.61	0.64
CrimeNet Explorer		0.47	0.43	0.45
LogAnalysis		0.60	0.45	0.51
SIIMCO	Betweennes Centrality	0.63	0.57	0.60
CrimeNet Explorer		0.43	0.33	0.37
LogAnalysis		0.49	0.42	0.45
SIIMCO	In Degree Centrality	0.76	0.72	0.74
CrimeNet Explorer		0.58	0.51	0.54
LogAnalysis		0.64	0.59	0.61
SIIMCO	Out Degree Centrality	0.81	0.83	0.82
CrimeNet Explorer		0.64	0.63	0.63
LogAnalysis		0.72	0.66	0.69

TABLE 3
PERFORMANCE OF THE THREE SYSTEMS ON THE DBLP DATASET BASED ON THE TOP VERTICES RETURNED BY THE STANDARD NETWORK METRICS

		Recall	Precision	F-value
SIIMCO	Closeness Centrality	0.53	0.56	0.54
CrimeNet Explorer		0.55	0.52	0.53
LogAnalysis		0.45	0.48	0.46
SIIMCO	Betweennes Centrality	0.59	0.47	0.52
CrimeNet Explorer		0.46	0.38	0.42
LogAnalysis		0.42	0.40	0.41
SIIMCO	In Degree Centrality	0.63	0.57	0.60
CrimeNet Explorer		0.49	0.43	0.46
LogAnalysis		0.57	0.52	0.54
SIIMCO	Out Degree Centrality	0.76	0.69	0.72
CrimeNet Explorer		0.61	0.53	0.57
LogAnalysis		0.65	0.57	0.61

3) Calculating the Euclidean Distances between the Results returned by the Systems and the Results determined by the Standard Network Metrics

We measured the average Euclidean Distance between the top *n* ranked vertices returned by each system and the corresponding top *n* ranked vertices returned by each of the four standard network metrics described previously. We considered *n* equaling 5, 10, and 15. We used the Euclidean distance measure shown in Equation 19.

$$d(\sigma_m,\sigma_s) = \sum_{x \in N_m^{top}} |\sigma_m(v) - \sigma_s(v)| \quad (19)$$

- N_m^{top} : List of the top n vertices returned by metric m
- $\sigma_m \in [0,1]^{|N_m^{top}|}$: Ranked list of the top n vertices returned by metric m.
- $\sigma_s \in [0,1]^{|N_m^{top}|}$: Ranked list of the top n vertices returned by system S.
- $\sigma_m(v)$ and $\sigma_s(v)$: Position of vertex $v \in N_m^{top}$ in the lists σ_m and σ_s respectively (a ranking of a set of n vertices is represented as a permutation of the integers 1, 2, . . . , n).

Fig. 5 shows the average Euclidean Distances using the three datasets. *Intuitively, the smaller the distance the better the system.*

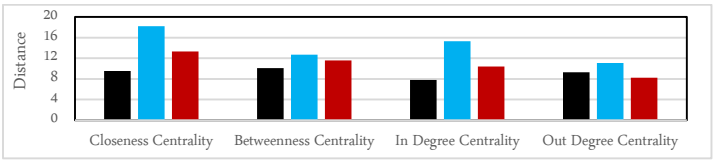
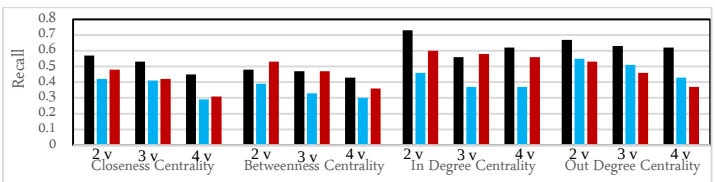


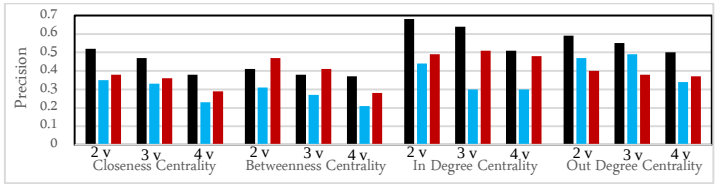
Fig. 5: The overall average Euclidean distances between the results returned by the three systems and the results determined by the four standard networks using Enron, Nodobo, and DBLP datasets.

C. Evaluating the Accuracy of the Three Systems to Identify the Top Influential Vertices of a Given List of Query Vertices

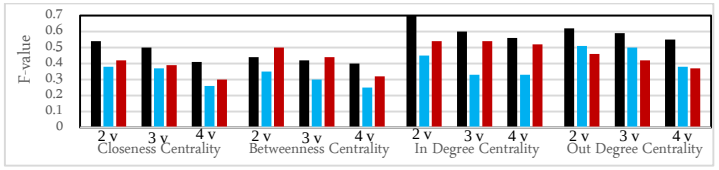
In this section, we evaluate the quality of the three systems to identify the immediate leaders of a given list of lower-level criminals in a criminal organization. Towards this, we evaluate the accuracy of the three systems to identify the most important vertices to a given list of vertices (i.e., query vertices) in the networks representing Enron and Nodobo datasets. We randomly selected 50 lists of 2-query vertices, 50 lists of 3-query vertices, and 50 lists of 4-query vertices from both Enron and Nodobo networks. We submitted the 150 sets of query vertices along with the networks representing the Enron and Nodobo datasets to the standard network metrics and the three systems. We considered only the top 5 vertices returned by each metric as the list l_{top} (recall section V-B-2). We compared the top 5 vertices returned by each system with the list l_{top}. We then calculated the Recall, Precision, and F-value of each system. Figs. 6 show the results for the Enron dataset. Figs. 7 show the results for the Nodobo dataset.



(a)

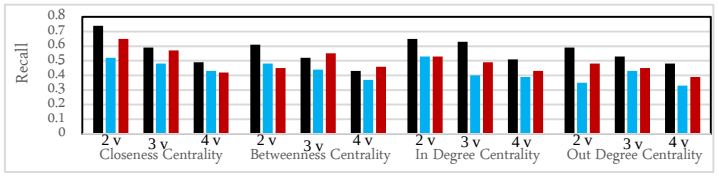


(b)

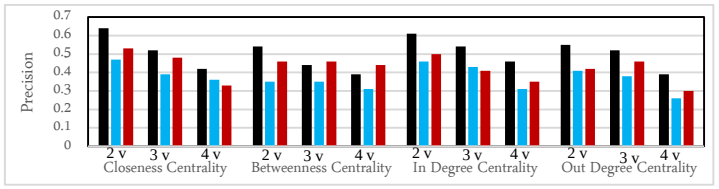


(c)

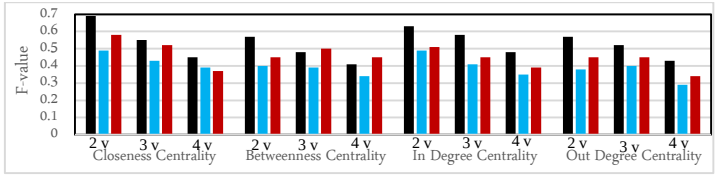
Figs. 6: (a) Recall, (b) Precision, and (c) F-value of the 3 systems for identifying the important vertices to a given list of query vertices on Enron dataset. 2v, 3v, and 4v denote 2 query vertices, 3 query vertices, and 4 query vertices respectively.



(a)



(b)



(c)

Figs. 7: (a) Recall, (b) Precision, and (c) F-value of the 3 systems for identifying the important vertices to a given list of query vertices on Nodobo dataset. 2v, 3v, and 4v denote 2 query vertices, 3 query vertices, and 4 query vertices respectively.

D. Discussion of the Results

As Figs. 3-7 and Tables 1-3 show, SIIMCO outperformed CrimeNet Explorer [22] and LogAnalysis [14]. The results revealed the robustness of the SIIMCO's method and its ability to identify the most important vertices in a network as well as the most important vertices to a given list of query vertices. Based on our observations of the results of the experiments, we attribute the performance of SIIMCO over CrimeNet Explorer and LogAnalysis to the following:

1. ***Incomplete contribution and inconsistent contribution:*** Due to the nature of the techniques employed by both CrimeNet Explorer [22] and LogAnalysis [14], some vertices may not contribute to the overall importance value of a vertex (Incomplete Contribution) and some vertices may contribute unequally to the overall importance value of a vertex (Inconsistent Contribution). These are due to the following:
    a) CrimeNet Explorer employs the well-known Degree, Closeness, and Betweenness Centrality metrics. Let v_x be the target vertex. Let v_y and v_z be vertices under consideration. The three metrics will assign a weight to

each of v_y and v_z based on the topology of each vertex in the network with regard to each of v_y and v_z. Therefore, v_y and v_z may have different weights. Consequently, their weight-based contributions to v_x may be unequal or they may not contribute at all to v_x.

b) LogAnalysis uses the Girvan and Newman algorithm [18] to compute edge betweenness. The algorithm considers the topology of each vertex in the network with regard to each vertex under consideration. Moreover, the algorithm uses a greedy strategy for partitioning the network into sub-communities by considering the topology of the entire network. Therefore, the contributions of vertices under consideration to a target vertex may be unequal or some of them may not contribute at all.

Let v be the current vertex under consideration. SIIMCO overcomes the problem of Incomplete Contribution by: (1) considering the importance of *each* query vertex to v, and (2) assigning a weight to each incoming edge to v that is outgoing from one of the query vertices (this weight represents the importance/rank of this vertex relative to all incoming edges to v). SIIMCO overcomes the problem of Inconsistent Contribution by: (1) considering the importance of *each* query vertex to each vertex connected to v, and (2) accounting for the degree of relativity of v to all query vertices. That is, SIIMCO overcomes the incomplete and inconsistent contribution limitations of most current methods outlined above.

2. ***Susceptibility to noise and outliers in data:*** CrimeNet Explorer and LogAnalysis are susceptible to noise and outliers in the data, from which a network is constructed. The weighting schemes adopted by SIIMCO shown in Equations 7, 10, and 13 make SIIMCO much less susceptible to noise and outliers in data.

3. ***Drawback of aggregating vertices and clusters hierarchically:*** The approach adopted by LogAnalysis that aggregates vertices and clusters a network hierarchically has the following limitations: (1) it is not suitable for clustering a large network, if some of the resulting sub-networks are large *(we observed that it clusters smaller subnetworks more accurately than larger sub-networks)*, (2) it is biased towards globular clusters, (3) it can never undo what was incorrectly grouped at an early phase, and (4) it may not handle different sized clusters accurately.

4. ***Limitation with the shortest path approach:*** To determine the degree of relationship between two vertices, CrimeNet Explorer uses the shortest path algorithm. The problem with this approach is that the weight of a vertex v is determined based only on the weight of the most important incoming edge e to v and the weight of the vertex connected to v by e. SIIMCO overcomes this problem by: (a) considering the weights of *all* incoming edges to v and the weights of *all* outgoing edges from v, and (2) considering the weights of all incoming and outgoing edges as a measure of importance of v. Towards this, SIIMCO assigns a weight to each incoming edge to v relative to all incoming edges to v; it also assigns a weight to each outgoing edge from v relative to all outgoing edges from v .

VI. Conclusion

We presented in this paper a forensic analysis system called SIIMCO. The proposed system can identify the influential members of a criminal organization and the immediate leaders of a given list of lower-level criminals. The techniques adopted by SIIMCO overcome the incomplete and inconsistent contribution limitations of most current methods. In the framework of SIIMCO, a network representing a criminal organization can be constructed from MCD that belongs to a criminal organization or from crime incident reports. In such a network, a vertex represents an individual criminal and a link represents the relationship between two criminals. SIIMCO adopts the concept space approach to construct a network automatically from crime incident reports. We constructed formulas that quantify the degree of influence of each criminal in a criminal organization relative to all other criminals in the organization. The formulas incorporate novel weighting schemes. We evaluated SIIMCO by comparing it experimentally with CrimeNet Explorer [22] and LogAnalysis [14]. For the evaluation, we used two real-world communication datasets: Enron email corpus [13, 16, 24] and Nodobo mobile phone records dataset [6, 35]. We also used DBLP dataset [12]. We evaluated the accuracy of the three systems by measuring their Recall, Precision, and Euclidean Distance with regards to: (1) ground truth data obtained from Enron's publicly known information, and (2) results determined by standard network metrics. Results revealed that SIIMCO outperforms the other two systems in terms of identifying the top influential vertices in a network and also identifying the top influential vertices to a given list of query vertices.

References

[1] A. Wesolowski, N. Eagle, A. J. Tatem, D. L. Smith, A. M. Noor, R. W. Snow, and C. O. Buckee, "Quantifying the impact of human mobility on malaria," *Science*, vol. 338, no. 6104, pp. 267–270, 2012.

[2] A. Milani Fard, M. Ester, Collaborative Mining in Multiple Social Networks Data for Criminal Group Discovery, IEEE International Conference on Social Computing (SocialCom), 2009.

[3] BREIGER, R. L. 2004. The analysis of social networks. In *Handbook of Data Analysis*, M. A. Hardy and A. Bryman, Eds. Sage Publications, London, U.K. 505–526.

[4] BREIGER, R. L., BOORMAN, S. A., AND ARABIE, P. 1975. An algorithm for clustering relational data, with applications to social network analysis and comparison with multidimensional scaling. *J. Math. Psych. 12*, 328–383.

[5] BAKER, W. E. AND FAULKNER R. R. 1993. The social organization of conspiracy: Illegal networks in the heavy electrical equipment industry. *Amer. Soc. Rev. 58*, 837–860.

[6] Bell, S., McDiarmid, A., Irvine, J. Nodobo: Mobile Phone as a Software Sensor for Social Network Research. Proceedings of Context Awareness for Proactive Systems, May 2011.

[7] Bower, K. When to Use Fisher's Exact Test. American Society for Quality, Six Sigma Forum Magazine, Vol. 2, No. 4, 2003.

[8] Baldi, P. & Hatfield, W. (2002), DNA Microarrays and Gene Expression, Cambridge University Press, Cambridge, UK. [80].

[9] CHEN, H. AND LYNCH, K. J. 1992. Automatic construction of networks of concepts characterizing document databases. *IEEE Trans. Syst. Man Cybernet. 22*, 885–902.

[10] CHEN, H., ZENG, D., ATABAKHSH, H., WYZGA, W., AND SCHROEDER, J. 2003. Coplink: Managing law enforcement data and knowledge. *Commun. ACM 46*, 28–34.

[11] Catanese, S., Ferrara, E., & Fiumara, G. (2013). Forensic analysis of phone call networks. Social Network Analysis and Mining, 3(1), 15-33.

[12] DBLP bibliography, 2014. [Online]. Available: http://www.informatik.uni-trier.de/ ley/db/

[13] Enron Corporation from Wikipedia. Available at: https://www.uwosh.edu/llce/conted/lir/course-listings/Enron%20Scandal.pdf.

[14] E. Ferrara, P. De Meo, S. Catanese, and G. Fiumara, "Detecting criminal organizations in mobile phone networks," *Expert Systems with Applications*, vol. 41, no. 13, pp. 5733–5750, 2014.

[15] EVAN, W. M. 1972. An organization-set model of interorganizational relations. In *Interorganizational Decision-Making*, M. Tuite, R. Chisholm, and M. Radnor, Eds. Aldine Publishers, Chicago, IL, 181–200.

[16] Enron Email Dataset. Available at: http://www-2.cs.cmu.edu/~enron/.

[17] F. Calabrese, M. Colonna, P. Lovisolo, D. Parata, and C. Ratti, "Realtime urban monitoring using cell phones: A case study in rome," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 1, pp. 141–151, 2011.

[18] Girvan, M., & Newman, M. (2002). Community structure in social and biological networks. Proceedings of the National Academy of Sciences, 99(12), 7821.

[19] Hauck, R. V., Atabakhsh, H., Ongvasith, P., Gupta, H., Chen, H. 2002. Using Coplink to analyze criminal-justice data. *IEEE Comput. 35*, 30–37.

[20] H. Sarvari, E. Abozinadah, A. Mbaziira, and D. McCoy, "Constructing and analyzing criminal networks," CA, USA, May 2014, pp. 84–91.

[21] H. Wang, C. K. Chang, H.-I. Yang, and Y. Chen, "Estimating the relative importance of nodes in social networks," *Journal of Information Processing*, vol. 21, no. 3, pp. 414–422, 2013.

[22] J. J. Xu and H. Chen, "CrimeNet explorer: A framework for criminal network knowledge discovery," *ACM Trans. Inf. Syst.*, vol. 23, no. 2, pp. 201–226, Apr. 2005.

[23] J. Pattillo, N. Youssef, and S. Butenko, "Clique relaxation models in social network analysis," in *Handbook of Optimization in Complex Networks*. Springer, 2012, pp. 143–162.

[24] Keila, P.S. and D.B. Skillicorn (2005), 'Structure in the Enron email dataset', Computational & Mathematical Organization Theory, 11(3), 183–99.

[25] Krebs, V. (2002). Mapping networks of terrorist cells. Connections, 24(3), 43–52.

[26] Kleinberg, Jon. Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM) 46 (5), 604-632.

[27] Klerks P., "The Network Paradigm Applied to Criminal Organisations: Theoretical nitpicking or a relevant doctrine for investigators? Recent developments in the Netherlands", Connections 24(3): 53-65, 2001, INSNA.

[28] LORRAIN, F. P. AND WHITE, H. C. 1971. Structural equivalence of individuals in social networks. *J. Math. Soc. 1*, 49–80.

[29] L. Cavique, A. B. Mendes, and J. M. Santos, "An algorithm to discover the k-clique cover in networks," in *Progress in Artificial Intelligence*. Springer, 2009, pp. 363–373.

[30] M. Akbas, R. Avula, M. Bassiouni, and D. Turgut, "Social network generation and friend ranking based on mobile phone data," 2013, pp. 1444–1448.

[31] MCANDREW, D. 1999. The structural analysis of criminal networks. In *The Social Psychology of Crime: Groups, Teams, and Networks*. D. Canter and L. Alison, Eds. Dartmouth Publishing, Aldershot, UK, 53–94.

[32] Meng Jiang, Peng Cui, Alex Beutel, Christos Faloutsos and Shiqiang Yang. CatchSync: Catching Synchronized Behavior in Large Directed Graphs. The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), New York City, USA, August 24 - August 27, 2014.

[33] Memon, Bisharat, *Identifying Important Nodes in Weighted Covert Networks Using Generalized Centrality Measures*. 2012 European Intelligence and Security Informatics Conference (EISIC 2012).

[34] Newman, M. (2004). Fast algorithm for detecting community structure in networks. Physical Review E, 69(6), 066133.

[35] Nodobo: Available at: http://nodobo.com/release.html

[36] P. I. Snchez, E. Mller, O. Irmler, and K. Bhm, "Local context selection for outlier ranking in graphs with multiple numeric node attributes," in *Proceedings of the 26th International Conference on Scientific and Statistical Database Management*, New York, NY, USA, 2014, pp. 1–12.

[37] Sageman, M. (2004). Understanding terror networks. University of Pennsylvania Press.

[38] Stanford Tokenizer, Part-of-Speech Tagger, and Named Entity Recognizer. Downloaded from: http://nlp.stanford.edu/software/

[39] Shang, X., Yuan, Y. *Social Network Analysis in Multiple Social Networks Data for Criminal Group Discovery*. 2012 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC).

[40] Taha, K. "Determining the Semantic Similarities among Gene Ontology Terms". *IEEE Journal of Biomedical and Health Informatics (IEEE J-BHI)*, 2013, Vol. 17, Issue 3, pp. 512 - 525.

[41] Todd, M., & Nomani, A. (2011). The truth left behind: Inside the kidnapping and murder of Daniel Pearl. New York. <http://www.publicintegrity.org/2011/01/20/2190/>.

[42] Taha, K., Homouz, D., Al Muhairi, H., and Al Mahmoud, Z. "GRank: A Middleware Search Engine for Ranking Genes by [502] Relevance to Given Genes". *BMC Bioinformatics* 2013, 14:251, doi:10.1186/1471-2105-14-251.

[43] Taha, K. and Elmasri, R. "SPGProfile: Speak Group Profile." *Information Systems (IS)*, 2010, Elsevier, Vol. 35, No. 7, pp. 774-790.

[44] Taha, K. and Elmasri, R. "BusSEngine: A Business Search Engine." *Knowledge and Information Systems: An International Journal (KAIS)*, 2010, LNCS, Springer, Vol. 23, No. 2, pp. 153-197.

[45] Taha, K. and Elmasri, R. "CXLEngine: A Comprehensive XML Loosely Structured Search Engine." *DataX'08 at EDBT'08 (Database technologies for handling XML information on the web), Nantes, France, March 2008.*

[46] U. K. Wiil, J. Gniadek, N. Memon; *Measuring Link Importance in Terrorist Networks. Social Network Analysis*, International Conference On Advances in Social Networks Analysis and Mining, ASONAM 2010.

[47] Wellman, B. 1988. Structural analysis: From method and metaphor to theory and substance. In *Social structures: A network approach*, B. Wellman and S. D. Berkowitz, Eds. Cambridge University Press, Cambridge, UK, 19–61.

[48] Yang, L. *Based on social network crime organization relation mining and central figure determining*. 2012 IEEE International Conference on Computer Science and Automation Engineering, June 2012.

Kamal Taha is an Assistant Professor in the Department of Electrical and Computer Engineering at Khalifa University, UAE, since 2010. He received his Ph.D. in Computer Science from the University of Texas at Arlington, USA, in March 2010. He has over 60 refereed publications that have appeared in prestigious top ranked journals, conference proceedings, and book chapters. Fifteen of his publications have appeared (or are forthcoming) in IEEE Transactions journals. He was as an Instructor of Computer Science at the University of Texas at Arlington, USA, from August 2008 to August 2010. He worked as Engineering Specialist for Seagate Technology, USA, from 1996 to 2005 *(Seagate is a leading computer disc drive manufacturer in the US)*. His research interests span Information Forensics & Security, bioinformatics, information retrieval, data mining, and databases, with an emphasis on making data retrieval and exploration in emerging applications more effective, efficient, and robust. He serves as a member of the Program Committee, editorial board, and review panel for a number of international conferences and journals, some of which are IEEE and ACM journals. He is a Senior Member of the IEEE.

Paul D. Yoo received his PhD in Engineering and IT from the University of Sydney (USyd) in 2008. He was a Research Fellow in the Centre for Distributed and High Performance Computing, at USyd from 2008 to 2009, and PHD Researcher (Quantitative Analysis) at the Capital Markets CRC, administered by the Australia Federal Dept. for Education, Science and Training, from 2004 to 2008. He was with the ATIC-Khalifa Semiconductor Research Center, KUSTAR from 2009 to 2014 as an Assistant Professor in Data Science. He is currently a Lecturer at the Data Science Institute, Bournemouth University, U.K. Paul also holds over 40 prestigious journal and conference publications and is currently actively involved in editorial board, technical program committees, and review panels of the data science and analytics areas for top conference and journal publications such as IEEE, ACM and ISCB. He is a Senior Member of IEEE.