Issues for the Next Generation of Criminal Network Investigation Tools

Uffe Kock Wiil
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
ukwil@mmmi.sdu.dk

Abstract—Criminal network investigation involves a number of complex knowledge management tasks and both humans and software tools play a central role in performing such tasks. The paper presents issues for future criminal network investigation tools. The research agenda is inspired partly by previous research from the hypertext field on how to build tools to structure, visualize, and manage knowledge and partly from previous research from the intelligence and security informatics field on tool support for criminal network investigation. CrimeFighter is used as an example to explore the limitations of current criminal network investigation tools and to propose issues that need to be addressed by future tools.

Keywords—research agenda, hypertext, intelligence and security informatics, CrimeFighter

I. INTRODUCTION

Criminal network investigation (CNI) involves a number of complex knowledge management tasks such as collection, processing, analysis, and visualization of information. Some of these tasks are best performed by humans and some tasks can benefit greatly from software tools to support (or automate) the involved workflow. Software tools outperform humans in collection, processing, and visualization tasks as the information volume and complexity increases, while human experience and (tacit) knowledge are needed for analysis tasks such as information synthesis and sense-making: “Modern technology has revolutionized intelligence data collection and processing; however, the man in the system is as critical as ever because only he can derive meaning from the data” [1]. The roles of humans and the roles of tools should be clearly identified and should be part of an overall scheme (model) governing the investigation process. Our approach to CNI is that the humans should always be in charge of all tasks in the investigation and that the tools are there to support the humans to do their job better and faster.

The proposed issues are inspired by previous work from both the hypertext and the intelligence and security informatics (ISI) fields. Hypertext systems have for more than three decades proven to be useful to support complex knowledge management tasks. For example, one of the early hypertext systems to support knowledge management was KMS [2]; KMS was used onboard the USS Carl Vinson. Tools for CNI have been discussed extensively in the ISI research literature. A number of CNI tasks that were previously conducted manually are now supported by software tools. These tools are becoming increasingly complex and provide support for a wide range of CNI tasks. However, these tools still have various limitations. CrimeFighter is used as an example to explore some of these limitations, and based on this the paper presents issues that need to be addressed by future CNI tools.

Section 2 provides an overview of previous and current tools for CNI, which divides the tools into three overall generations. Section 3 describes the particular perspective that the author has on CNI tools which is partly inspired by previous work from the hypertext field. An overall process model for CNI is also presented. CrimeFighter is briefly presented as an example of a third generation CNI tool. In Section 4, we present issues that are considered to be critical; these issues are exemplified by limitations in CrimeFighter. These issues need to be thoroughly examined in future research and become part of future CNI tools. Section 5 concludes the paper and outlines future work.

II. CRIMINAL NETWORK INVESTIGATION TOOLS

Criminal network analysis and visualization tools have previously been categorized into three overall generations. Klerks [3] was the first to do so in 2001; many others followed later (e.g., [4, 5, 6]).

The first generation tools take a manual approach. The Anacapa chart [7] is a representative of this generation. An investigator first constructs an association matrix manually from raw data by identifying criminal associations. A link chart can then be drawn manually based on the association matrix for visualization purposes. This enables the investigator to study the network to try to discover patterns of interest. For example, Krebs [8] used this approach to map the 9/11 terrorist network consisting of the 19 hijackers. Although such a manual approach is helpful for CNI, it becomes increasing ineffective and inefficient as the data set increases in size. Sparrow [9] labeled Anacapa charts as the state-of-the-art in law enforcement uses of network analysis (in 1991). Sparrow also suggested the use of computer tools with graphic display capabilities to do more advanced link and network analysis – including the use of concepts from social network analysis (SNA) [10].

Second generation tools is characterized by their graphic-based approach. Second generation tools are more
sophisticated since they can automatically produce graphical representations of criminal networks. Tools in this generation include Analyst’s Notebook [11], Netmap [12], Xanalys Link Explorer (previously called Watson) [13], and Coplink [14]. These tools provide various levels of interaction and pattern identification, representing information using visual clues and algorithms to help the user understand charted relationship [6]. Although, second generation tools are capable of using various methods of visualizing criminal networks, their sophistication level remain modest because they offer little structural analysis capability [4]. Such tools still rely primarily on the investigator to do structural analysis.

Third generation tools possess advanced structural analysis capability. Various advanced analysis features are found in tools belonging to this generation including features from SNA (e.g., [9, 10]), Dynamic Network Analysis (DNA)\(^2\) (e.g., [15, 16]), Terrorist Network Analysis (TNA)\(^3\) (e.g., [17, 18]) as well as features from Semantic Networks (SN) [19, 20]. Tools belonging to this generation provide advanced analytical facilities that can help the investigator to (1) discover characteristics of criminal networks such as central members, network roles, central relations, subgroups, hidden hierarchy, interaction patterns between groups, overall network models, etc., (2) provide capabilities to predict missing members, relations, subgroups, etc., (3) track, display, and forecast network evolution – and so on. The list is growing and new advanced capabilities are regularly proposed in the ISI research literature. Example tools belonging to this generation include ORA [21], Palantir Intelligence [22], Dark Web [23], and Dynalink [24] as well as CrimeFighter [25, 26].

One could argue that the classification into second and third generation tools have become somewhat fuzzy. Some tools that have originally been labeled as belonging to the second generation (e.g., Coplink, Analyst’s Notebook, and Xanalys Link Explorer) have evolved over time and now include features labeled as belonging to third generation tools. However, the idea of thinking in terms of generations of CNI tools with increasingly advanced features fits well with the idea and message of this paper.

III. PERSPECTIVE

Investigative models typically include the following overall knowledge management processes: acquiring the needed information (collection and processing), creating a model of the target (synthesis), extracting useful information from the model (sense-making), and creating a representation of the results (dissemination). Based on a specific target-centric model for intelligence analysis [27], we have previously proposed a generic model for target-centric investigative teams [28] (see Figure 1). The investigation model was developed to embrace three specific types of investigations: policing [7, 29, 30], counterterrorism [27, 31, 32], and investigative journalism [33, 34, 35].

The workflow of the model is as follows [28]: The customer requests information about a specific target. The investigators request information from the collectors (that may also be investigators). Information related to the target is acquired in disparate pieces over time. The investigators use the acquired information to build a model of the target (synthesis) and extract useful information from the model (sense-making). The extracted information results in changes to the model (synthesis). The sense-making – synthesis cycle is continued throughout the investigation as new information is acquired and extracted from the model. The investigators both work individually and cooperatively as a team. The results of the investigation are disseminated to the customer at the end of the investigation or at certain intervals (or deadlines).

Investigation is a human-centered knowledge management process. Investigators (and collectors) rely heavily on their past experience (tacit knowledge) when conducting investigations. Hence, these processes cannot be fully automated and taken over by software tools. The overall philosophy of CrimeFighter is that the humans (in this case the investigators) are in charge of the investigative tasks and the software tools are there to support them. The tools should be controlled by the investigators and should support (1) complex intellectual tasks (e.g., synthesis and sense-making), (2) time-consuming tasks (e.g., collection, processing, and dissemination), and (3) cooperative tasks to allow the investigators to reach better results faster. This perspective places some tough requirements on the tools. When investigators use software tools to (semi) automate previous manual tasks then transparency becomes an important issue. Investigators need to know how the tools work to be able to trust the results they generate.

CrimeFighter can be classified as belonging to the third generation of CNI tools; it consists of three tools that each focus on supporting one or more tasks of the overall process model. CrimeFighter Explorer focuses on providing support for collection and processing tasks, CrimeFighter Assistant focuses on providing support for sense-making tasks (network analysis and visualization), while CrimeFighter Investigator focuses on providing support for acquisition, synthesis, sense-making, and dissemination tasks. Support for cooperation beyond simple information sharing (import and export of data sets) is planned for future versions of the tools. Due to the particular focus of this paper on CNI, only the features of CrimeFighter Assistant and CrimeFighter Investigator will be described.

\(^2\) The term DNA first appeared in print in a paper published by Carley [15].

\(^3\) The term TNA has been used to describe analysis features specifically tailored for terrorist networks as seen as a specialization of social networks.
A. CrimeFighter Assistant

In essence, CrimeFighter Assistant [18, 26] provides support for some “typical” SNA features, some “specialized” TNA features as well as network visualization. Figure 2 shows the tool interface consisting of the visualization view to the left and the analysis results view to the right.

The starting point for the tool is the existence of a terrorist (criminal) network. The tool has been developed to assist an investigator in gaining better understanding of the overall network as well as its parts (nodes and links).

The tool can help in answering various questions such as (1) Network: How covert is the network? How efficient is the network? What is the density of the network? What is the trade-off between secrecy and efficiency in the network? (2) Nodes: Who are the central (important) persons in the network? What makes the person important? What role does a particular person have? How is the network affected after removal of a particular node? (3) Links: What links are important for communication in the network? How important is a particular link in relation to network efficiency and secrecy? What is the information backbone of the network? Hence, the available measures have been divided into the same three categories: network, nodes, and links.

Currently, the following network measures are supported: density, diameter, secrecy [17], efficiency [36] and performance [17]. The first two are SNA measures [10], while the latter three are TNA measures. The following node measures are supported: node degree centrality, node closeness centrality, node betweenness centrality, node eigenvector centrality, and position role index. The first four are SNA measures, while the latter is a TNA measure. The following link measures are supported: link betweenness [37] and link importance [18]. The first is a SNA measure, while the latter has been developed as a TNA measure.

B. CrimeFighter Investigator

In this brief description, the focus will be on the support for synthesis and sense-making. Additional details about CrimeFighter Investigator can be found in [25, 28, 38]. Figure 3 displays some important concepts of the tool.

At the center (Figure 3) is the shared information space. Spatial hypertext research has inspired the features of the shared information space including the support of investigation history and the structural parser [39]. The view concept provides investigators with different perspectives on the information in the space and provides alternative interaction options with information (hierarchical view to the top left; satellite view to the bottom left; spatial view at the center; algorithm output view to the right). Finally, a structural parser assists the investigators by relating otherwise unrelated information in different ways, either based on the entities themselves or by applying algorithms to analyze them.

Synthesis support assists investigators in enhancing the target model. Central synthesis support includes: (1) Creating, editing, and deleting entities. Investigators think in terms of people, places, things, and their relationships. (2) Creating, editing, and deleting relations. The impact of link analysis on investigative tasks is crucial to the creation of the target model. (3) Re-structuring. Investigation information structures are typically emerging and evolving, requiring continuous restructuring of entities and relations. (4) Grouping. Investigators often group entities using symbols like color and co-location (weak), or they use labeled boxes (strong). (5) Collapsing and expanding information is essential since the space available for manipulating information is limited physically, perceptually, and cognitively.

Sense-making support assists investigators in extracting useful information from the synthesized target model. Central sense-making support includes: (1) Generating hypotheses and competing hypotheses is a core task of investigation that involves making claims and finding supporting and opposing evidence. Investigators often retrace the steps of their investigation to see what might have been missed to evolve an existing hypothesis or start a new one. (2) Adaptive modeling. Representing the expected structure of networks for pattern and missing information entity detection is a proactive sense-making task. Adaptive modeling embeds the tacit knowledge of investigators in network models for prediction and analysis. (3) Prediction. The ability to determine the presence or absence of relationships between and groupings of people, places, and other entity types is invaluable when investigating a case. (4) Exploring perspectives. To reduce the cognitive biases associated with a particular mind set, exploring different perspectives (views) of the information is a key CNI task. (5) Decision-making. During an investigation, decisions have to be made such as selecting among competing hypotheses.
IV. ISSUES

A criminal network is as the term indicates based on a network structure. Previous hypertext research can be used as an important source of inspiration for criminal networks and tools supporting CNI. Hypertext research has for more than three decades focused on the use of different models for structuring and manipulating information – including network models (navigational hypertext e.g., [40]), tree (hierarchy) models (taxonomic hypertext e.g., [41]), semantic network models (issue-based hypertext e.g., [42]), and spatial proximity models (spatial hypertext e.g., [43]) as the most prominent ones. Numerous systems have been developed to explore and support various aspects of these types of hypertext structures. In 1988, Halasz published a paper describing seven issues that needed to be addressed by future hypertext systems [44]. These issues are very generic in nature and still inspire much work in the hypertext field.

This paper discusses the original issues raised by Halasz as well as issues that we have encountered in our own work – all of them placed in a modern context of tool support for CNI. The issues are divided into four overall categories: data model, computational engine, advanced functionality, and system features. The issues are not ordered in terms of priority or importance.

A. Data Model

The data model is central in the sense that it enables the modeling and network measures capability of the tool. If the data model becomes too simple, it will not be able to model the complex real world networks related to terrorism, organized crime, etc. Also, a simple data model will limit the possibilities for computing metrics about the network.

In relation to CrimeFighter, we studied various application domains that use network structures as a modeling concept; we also looked at various related work to get a feeling for the state-of-the-art. We noticed that many tools relied on data models with only nodes (vertices) as first class entities. The links (relations or edges) between nodes were often stored as properties of the nodes. We also noticed that most network measures either focus on the overall network (e.g., density and diameter) or the nodes (e.g., various node centralities). The only link measure that we came across was link betweenness [37].

In CrimeFighter Assistant, links became first class entities in the same way as nodes are (with the ability to have attribute/value pairs attached to them). We also developed a new link measure (link importance [18]) to try to promote the “value” of the link in network measures. However, there are still many issues that need to be addressed.

Link as first class entity. To increase the modeling capability to match real world scenarios, links should have a direction (e.g., “to”, “from”, “bi-direct”, and “none”) and a weight (e.g., a decimal value in the range [0..1] or an integer value greater than or equal to zero). Adding link directions and weights will require changes to the way that network measures are computed. Hence, it should be considered how the various link attributes should be included in the measures. For example, should node degree centrality rely on the total number of links that connects the node, on the total weight of links that connects the node, or on a combination of both? How should the link direction influence the node degree centrality? Does an incoming link count for more than an outgoing link? Questions like these needs to be answered for all network measures to fully incorporate the link as a first class object in the network model. Recently, we did a first study of how link weights can be incorporated into node centrality measures [45] based on the work of Opsahl et al. [46].

Typed nodes and links. Nodes and links should be typed (as known from semantic networks) to be able to model complex covert networks that consist of various types of nodes (e.g., people, places, events, etc.) and various types of links (e.g., friend of, lives in, participated in, etc.). A network of typed nodes and links require changes to how network measures are calculated. For example, most node centrality measures only work for single-modal, single-relational networks (i.e., a network where all nodes are of the same type and all links are of the same type). Hence, network measures need to be generalized to take into account that nodes and links can be of different types. The problem is as following: given a network of typed nodes and links and the desire to calculate the node degree centrality to find the most central person in the network, one needs to prune the multi-dimensional network to reduce the “dimensionality” of the network. Much information about the network is lost in the pruning process. How reliable will the measure then be? We have just recently begun looking into analysis and visualization of what we call multi-modal, multi-relational, and multi-featured (3M) networks [47] – networks with multiple types of nodes, links, and attributes (features).

Composites. When studying criminal networks, groups play an important role in the understanding of such networks. Hence, the network model should be augmented with another first class modeling concept (composites). A composite is a grouping mechanism that can group nodes and links into a higher level structure. Like nodes and links, composites can be typed and have attribute/value pairs attached.

In CrimeFighter Investigator composites indicate subgroups of an overall network. Composites group nodes by reference (as opposed to inclusion). An inclusion relation implies a part/whole relationship in which characteristics of and operations on the whole also will affect its parts as well [44]. Reference implies a much looser relationship in which the participating entities allude to each other but remain essentially independent [44]. Link composite is another type of composite in CrimeFighter Investigator that allows investigators to group multiple links between two nodes (such as multiple emails or phone calls between two persons) into a single visible (composite) entity. Link composites group links by inclusion. A third type of composite in CrimeFighter supports collapsing and expanding. This type of composite groups all nodes by inclusion. Links that are internal to (have both endpoints inside) the composite are also included, while external links (one endpoint outside) are referenced. This type of composite supports the concept of a subspace that allows the investigators to work in detail with a portion of the overall network. A subspace provides the same functionality as the main space.
The composite concept is a powerful modeling concept and much work has been done in the hypertext community to explore various issues related to this concept [48] in relation to the Dexter hypertext reference model. CrimeFighter Investigator made a first limited effort in adopting the concept to the criminal networks domain. However, there are still many issues that need to be explored. For example, what types of composites should be available? For each type, the reference versus inclusion issue much be addressed. Can a composite group other composites? Can a node or a link be included in more than one composite? How should the concept be visualized in the user interface as an integral part of the network structure? To what extent can the concept be used to support other features of CNI – such as virtual structures and versioning?

B. Computational Engine

The computational engine is the term used to describe the overall functionality of a CNI tool that is needed to enable the desired features related to analysis and visualization to be added to the tool.

Typical CNI features that current CNI tools support are SNA, TNA, subgroup detection, hierarchy detection, community detection, etc. An important issue in this regard (as mentioned above) is the necessary generalization of these features in relation to a more powerful data model with nodes, links, and composite as first class entities. CrimeFighter Assistant and CrimeFighter Investigator both support a subset of these traditional features. New features are frequently reported in the research literature. Hence, a CNI tool should be able to evolve over time to include the relevant features; the computational engine should allow new features (algorithms) to be added as well as existing features to be updated or removed. Hence, an important question is how to support this kind of “plug and play” of computational features?

Prediction. Predictive features such as extrapolation, projection, and forecasting have been around in many disciplines for some time [27, 50]. Naturally, they have also found their way into CNI. “In intelligence analysis, we are concerned with describing the past and current state of the target to make a prediction about its future state”. [27] CrimeFighter Investigator has made a first initial step in this direction by using an inference-based predictive technique to support detection of missing links as part of a workflow of node removal [51]; the work is based on prediction techniques developed by Rhodes [52]. Prediction based on different entities (node, link, and composites) and their attributes should be supported in order to help determine the presence or absence of relationships between and groupings of people, places, and other node types. Clark describes several prediction techniques that can be used for these purposes [27]. Important questions are: how CNI tools can best support these techniques? And, what techniques it makes sense to support in order to optimize tool support for CNI? The possibilities in this area seem virtually unlimited.

Search and query. Most CNI tools provide some kind of graphical overview of the network under investigation (as do both CrimeFighter Assistant and CrimeFighter Investigator). This kind of navigation support (overview) is important to gain understanding of the network, but it is not sufficient in itself. Effective access to information about the network requires a query-based approach to compliment the network overview. There are two broad types of query mechanisms that can play an important role in accessing information: content search and structure search [44]. In content search, all entities in the network are examined individually for a match to a given query (e.g., find all nodes in the network that contain an attribute named “place” with the value “Odense”). Hence, context search ignores the structure of the network. In contrast, structure search specifically examines the structure of the network for a match to a given pattern or substructure (e.g., find all substructures where two nodes of type “person” are connected by a link of type “friend”).

The next step is to define query languages for both types of queries. It would also make sense to consider how structure and content queries can be combined to match complex patterns in the criminal network (i.e., complex queries based on both structural aspects and properties/attributes of entities). An important consideration is how to express such queries. It seems that structural queries would best be expressed in some sort of graphical editor.

Virtual structure. Criminal networks are essentially dynamic in nature. They change over time in discrete steps. Virtual structure is a way to support dynamic visualization of the network data through the concept of views. Views are already a well-known feature in CNI tools (including CrimeFighter) and many types of views exists – including network (link chart), temporal (timeline), (geo)-spatial (map), hierarchy (org chart), etc.

The concept of virtual structure relies heavily on the search and query capabilities of the CNI tool. An entity in a virtual structure is in fact a query. The concept of virtual structure is also tightly connected to the concept of composites. A virtual composite allows the user to specify views of the network data that are created at runtime by evaluating a query. The concept of a virtual link is also intriguing. A virtual link could for instance have a known (specified) node as one endpoint and a query as the other endpoint. Hence, the second endpoint is “calculated” at runtime when the view is opened. The concept of virtual links would work well with a versioning mechanism. For example, a link would be able to always point to the latest version of a node. A virtual node could likewise be based on a query. Hence, the content of the node would be created at runtime based on a query. An interesting issue to think of is whether it would make sense to store virtual structures (in particular generated views) for later use? This could be relevant in relation to documentation and dissemination aspects of CNI. Also, the relation between virtual structures and versioning is crucial to investigate further. Implementing virtual structures (composites, links, and nodes) will be a challenging task for the next generation of CNI tools. However, the potential benefits are exciting. Virtual structures add a whole new “dynamic” dimension to the concept of views.
C. Advanced Functionality

Many types of advanced functionality could be envisioned for future CNI tools. Two specific types will be discussed here: versioning and cooperation support.

Versioning is an important feature in relation to criminal networks. A good versioning mechanism will allow users to maintain and manipulate a history of changes to their network [44]. CrimeFighter Investigator provides a history mechanism that supports users with unlimited undo/redo of actions in the shared information space. However, the history mechanism does not support branched history (i.e., the ability to simultaneously explore more than one alternative structure configuration of the information).

Providing a branched version history for all entities in a criminal network raises some very difficult issues. Can composites be used as version sets encapsulating the version history of a subset of the network? What underlying changes (granularity) to nodes, links, and attributes should trigger a new version of the composite? If you link to a versioned node, what version will you then get? The latest version, a specific version (i.e., the one active at the time of the creation of the link, or a version selected based on a query)? There are many, many such design decisions that needs to be made to develop a powerful versioning mechanisms that works at all information levels (composite, node, link, and attribute) in a criminal network. However, if such a mechanism were in place, it would be quite useful.

Cooperation support beyond simple data sharing is not really supported in the current generation of CNI tools. For example, a user can work on a criminal network, store it, and send it someone else that can work on it afterwards. This type of asynchronous cooperation is still useful, but one could easily envision work scenarios where synchronous cooperation of multiple investigators would be beneficial [31, 32]. “Software tools that make it easier for analysts to collaborate as a natural part of their work will lead to better analysis that is informed by more perspectives” [53].

The next generation of CNI tools should improve the support for seamless cooperation – including support for the overall types of cooperative work settings (see Figure 4). Moving from essentially single user systems to multi user cooperative systems require support for a number of additional mechanisms in future CNI tools. Mechanisms such as access control, locking (concurrency control), notification, and transaction become vital (e.g., [55, 56]). Again, important design decisions need to be made regarding the granularity of these mechanisms. Should they only work at the composite level, node/link level, or at the attribute level? Also, many issues and concepts must be taken into account in cooperative CNI tools to support the dynamics and coordination aspects of a multi user interface [54].

D. System Features

This category deals with issues that relate to the overall system (or tool) or the practical use of such tools.

Extensibility and tailorability. As can be seen in the above discussion of issues, the ability to extend and tailor the various parts of the tool seems important.

At the data model level, it is important to be able to adapt the data model to the investigation at hand. This includes the ability to define new entity types and to include attribute/value pairs in all types of entities to accommodate the necessary modeling power of the particular type of criminal network being investigated.

With respect to the computational engine, it is important to be able to add new computational features (algorithms) and to adjust the ones available to fine-tune them to address the particular challenges and needs of the investigation at hand. As mentioned above, a plug and play like approach to manipulating computational features seems feasible.

Regarding the new mechanisms that must be added to address the advanced functionality related to versioning and cooperation support, previous work [55] indicated that a particular approach that thinks in terms of separating the actual mechanism, the policy, and the presentation is helpful. For example, regarding a notification mechanism, one would need to develop generic, fine-grained, and broadly applicable functionality for the actual mechanism. Then, one would need to decide on the policy or how the mechanics should be deployed in the particular tool (i.e., what types of actions by one user should trigger notifications to other users?). Finally, one would need to decide how notifications are incorporated into the user interface (i.e., should an update notification generated by actions of another user related to an active entity automatically result in the entity being updated on the screen?). Such separation on concerns would allow for maximum flexibility in the development of CNI tools.

Extensibility and tailorability can be achieved in different ways. Previous work (e.g. [55]) has evolved much around the use of extensible programming languages/environments (such as Lisp and Scheme) as well as object-oriented languages (in particular Java). These languages all have the ability to be extended at runtime with additional features.

An important issue to keep in mind with respect to extensibility and tailorable is the skill level of the investigator. CNI tools should be useful more or less as is for normal computer literate investigators as well as provide means for extensibility and tailorable for highly computer literate investigators (“super users”).

![Figure 4. Time space taxonomy [54].](Image)
Openness. When developing CNI tools, it is important to think in terms of openness. Tools can be open in various ways. CrimeFighter tools use GraphML [57] as storage and interchange format. This allows data sets to be shared among tools. Various other similar formats exist (e.g., DyNetML [58]). However, the ISI community should aim for an increased level of openness and interoperability.

The hypertext community has over the years developed community reference models to address the issues of openness. The Dexter hypertext reference model (see above) is one such approach [49]. Another approach is the work of the Open Hypermedia Systems Working Group (OHSWG) that focused on developing de facto standards for the community. A protocol was developed that allowed open hypermedia systems to interoperate at runtime – exchanging data between systems and creating network structures that spanned multiple systems developed by independent research groups [59]. Both these efforts made a huge impact on subsequent work in the hypertext community.

It would be relevant and interesting to think in terms of a common reference model for CNI tools developed by the ISI research community as a joint effort to the benefit of the community. Such an effort should think in terms of definition of an overall CNI tool (reference) architecture consisting of a set of components with well-defined interfaces (APIs). This would allow for an increased level of openness and interoperability where components of one CNI tool can use (be based on) components of other CNI tools. Such a community effort would result in strong ties between core research groups in the field and would eventually result in better tools that can be developed faster through the community standards and tools. Openness also implies open source. Maybe the result of a joint community effort could be an open source project that allows scholars from all over the world to contribute to parts (components) of the overall tool (as defined in the reference architecture). This would really take the ISI community to the next level.

Scalability. Of course, scalability is important when discussing CNI. As research prototypes, CrimeFighter tools have only been deployed with data sets of thousands of entities. Hence, we do not know how well they scale to larger data sets of millions of entities. One reason for this is the lack of large open source data sets. Hence, it would be interesting to have one or more ISI community reference data sets that can be used for such purposes – both in relation to experiments related to scale but also in relation to other metrics (e.g., machine learning and algorithm performance). This would allow researchers to better compare their results with others and position themselves in relation to previous work in the field.

The primary contribution of the paper is the set of issues. Due to space limitations, the issues are not discussed at a high level of detail. A trade-off was made to include more issues at the expense of providing additional detail about each issue. It is the hope that the provided references will provide additional inspiration on how to address the issues.

No doubt, some issues have already been explored by researchers to some extent. However, there are still much to be done related to each issue. Hence, the claim is that none of these issues have been fully explored.

Future work plans evolve around addressing the above issues in relation to CrimeFighter CNI tools, issue by issue, over the years to come. Currently, various projects are underway, each trying to address different issues.

ACKNOWLEDGMENT

The CrimeFighter project started in 2009. The work presented in this paper is much inspired by past interactions with CrimeFighter team members – in particular Jolanta Gniadek (M.Sc. 2010), Rasmus R. Petersen (Ph.D. 2012), and Bisharat R. Memon (Ph.D. student).

REFERENCES


