Concept Matching in Informal Node-Link Knowledge Representations

by

Byron Bennett Marshall

A Dissertation Submitted to the Faculty of the

COMMITTEE ON BUSINESS ADMINISTRATION

In Partial Fulfillment of the Requirements
For the Degree of

DOCTOR OF PHILOSOPHY
WITH A MAJOR IN MANAGEMENT

In the Graduate College

THE UNIVERSITY OF ARIZONA

2005

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Dissertation Committee, we certify that we have read the dissertation prepared by Byron Bennett Marshall

entitled Concept Matching in Informal Node-Link Knowledge Representations

and recommend that it be accepted as fulfilling the dissertation requirement for the Degree of Doctor of Philosophy

_____ Date: 04/29/2005
Hsinchun Chen

_____ Date: 04/29/2005
Mohan Tanniru

_____ Date: 04/29/2005
Therani Madhusudan

_____ Date: 04/29/2005
Terry Langendoen

Final approval and acceptance of this dissertation is contingent upon the candidate's submission of the final copies of the dissertation to the Graduate College.

I hereby certify that I have read this dissertation prepared under my direction and recommend that it be accepted as fulfilling the dissertation requirement.

_____ Date: 04/29/2005
Dissertation Director: Hsinchun Chen

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: _Byron Bennett Marshall_

ACKNOWLEDGEMENTS

I thank Dr. Hsinchun Chen for his guidance and support as I developed and pursued this work. Various members of the faculty of the MIS department have also contributed to my work and academic development. Thank you to Dr. Therani Madhusudan, Dr. Mohan Tanniru, and Dr. Terry Langendoen who are part of my committee, Dr. Jay Nunamaker and Dr. Kurt Fenstermacher who have taken extra time to chat with me and advise me on numerous occasions, and Dr. Leon Zhao who helped me with various administrative hurdles as a member of the departmental graduate committee. I am deeply indebted to these and other MIS faculty members. Thanks also to Cathy, Dan, Hua, my co-authors, and my other colleagues in the AI Lab and in the department who have been (and will continue to be) my research partners and friends. This work has been supported by research grants from DHS, NSF, and NLM/NIH. A special note of thanks also goes to Allan Knepper who so substantially impacted my professional and personal development before and during my time at Dunavant of California. Most of all I am grateful for the constant support and understanding of Sarah Marshall and the rest of my family without whom none of this work would have been possible.

TABLE OF CONTENTS

TABLE OF CONTENTS – *Continued*

TABLE OF CONTENTS – *Continued*

LIST OF ILLUSTRATIONS

LIST OF TABLES

ABSTRACT

Information stored by managed organizations in free text documents, databases, and engineered knowledge repositories can often be processed as networks of conceptual nodes and relational links (concept graphs). However, these models tend to be informal as related to new or multi-source tasks. This work contributes to the understanding of techniques for matching knowledge elements: in informal node-link knowledge representations, drawn from existing data resources, to support user-guided analysis. Its guiding focus is the creation of tools that compare, retrieve, and merge existing information resources.

Three essays explore important algorithmic and heuristic elements needed to leverage concept graphs in real-world applications. Section 2 documents an algorithm which identifies likely matches between student and instructor concept maps aiming to support semi-automatic matching and scoring for both classroom and unsupervised environments. The knowledge-anchoring, similarity flooding algorithm significantly improves on term-based matching by leveraging map structure and also has potential as a methodology for combining other informal, human-created knowledge representations. Section 3 describes a decompositional tagging approach to organizing (aggregating) automatically extracted biomedical pathway relations. We propose a five-level aggregation strategy for extracted relations and measure the effectiveness of the BioAggregate tagger in preparing extracted information for analysis and visualization. Section 4 evaluates an importance flooding algorithm designed to assist law enforcement investigators in identifying useful investigational leads. While association networks have

a long history as an investigational tool, more systematic processes are needed to guide development of high volume cross-jurisdictional data sharing initiatives. We test path-based selection heuristics and importance flooding to improve on traditional association-closeness methodologies.

Together, these essays demonstrate how structural and semantic information can be processed in parallel to effectively leverage ambiguous network representations of data. Also, they show that real applications can be addressed by processing available data using an informal concept graph paradigm. This approach and these techniques are potentially useful for workflow systems, business intelligence analysis, and other knowledge management applications where information can be represented in an informal conceptual network and that information needs to be analyzed and converted into actionable, communicable human knowledge.

1 INTRODUCTION

1.1 Transforming Data into Knowledge

"Where is the wisdom we have lost in knowledge?

Where is the knowledge we have lost in information?"

These words from T.S. Eliot's play The Rock (Eliot, 1934) p.7 are perhaps, the original source of the familiar DIKW (Data-Information-Knowledge-Wisdom) hierarchy often cited in studies of organizational knowledge management (Cleveland, 1982). Although (Zeleny, 1987) maps data to "know-nothing," information to "know-how," knowledge to "know-what", and wisdom to "know-why," Russell L. Ackoff's 1988 presidential address to ISGSR (Ackoff, 1989) is widely considered the first mention of the DIKW hierarchy in the knowledge management (KM) literature (Sharma, 2005). In any case, IT practitioners generally agree that there exists a continuum of data, information, and knowledge within any enterprise (Chen, 2001). Although transforming data into information and information into knowledge is a well recognized organizational knowledge management process, T. S. Eliot's lament might be echoed by human analysts who are frequently lost in vast amounts of available data and information.

1.2 Leveraging Organizational Knowledge Assets

While knowledge may be best defined as something which exists only in the minds of people, it is clear that organizations are well acquainted with the need to translate data ("discrete, objective facts about events") into information (a "message")

and then into knowledge ("a fluid mix of framed experience, values, contextual information, and expert insight") (Alavi & Leidner, 2001; Davenport & Prusak, 2000). A wide variety of technological elements are also used by organizations to manage knowledge assets. Computer technology has enabled organizations to generate and store vast collections of data in databases and information in free text documents such as reports and email messages. In addition, vast collections of digital resources are available in digital libraries and on the Internet. Knowledge bases, ontologies, taxonomies and thesauri are frequently developed to enhance the usefulness of these important resources. The usefulness of an organization's knowledge management technology is demonstrated when available knowledge assets are organized and presented to support decision making, analysis, or other organizational tasks.

Figure 1.1 depicts the process of matching a user's query to knowledge elements stored in knowledge resources. In the Figure, a user/analyst forms a query consisting of a question in the context of some known information. An analysis application interprets the query using some stored information models to locate appropriate information in a set of data/information resources. The identified knowledge elements are then returned to the human analyst. Data models can include hierarchies, ontologies, knowledge bases, and other kinds of indexes. Retrieved information consists of information generated by "authors" who express what they "know" in a model of some form. Examples of data/information resources include databases such as work flow repositories and law enforcement records, conceptual structures such as concept maps, as well as less structured textual resources like articles, reports, and web pages.

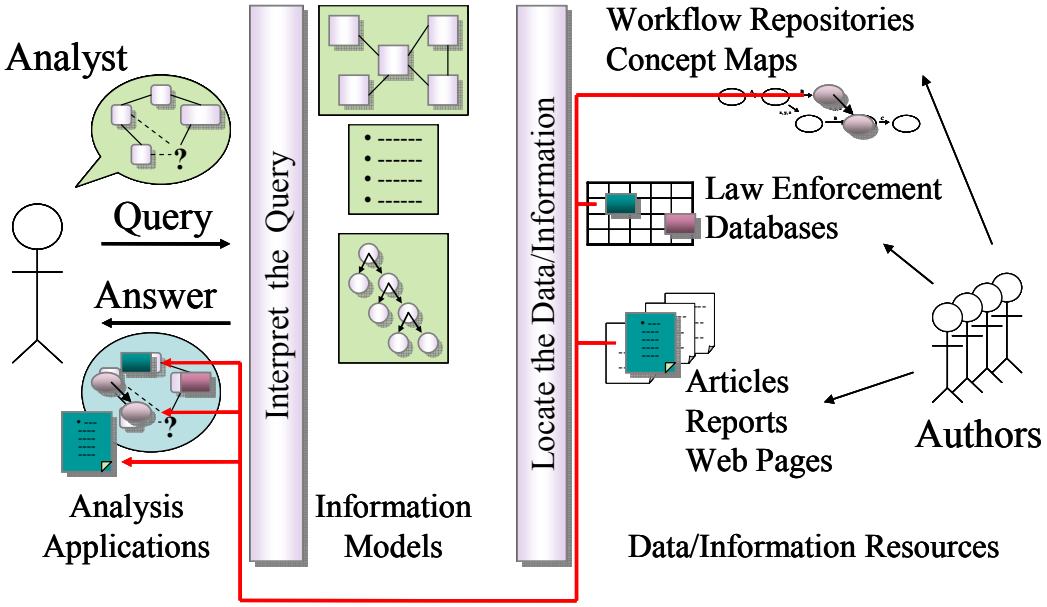


Figure 1.1. Information Models Connect Users to Resources

To systematically address organizational analysis processes with technology, we need to consider the structure and characteristics of available data and information sources. Organizational data and information elements come in many forms. Figure 1.2 depicts information captured in different repositories and emphasizes the need to match conceptual elements. The analyst's query is handled by matching conceptual elements in existing resources that may or may not have been constructed to support the specific query task. Resources include free text (shown in green) and more structured databases (rose). Access to these resources may be supported by carefully constructed knowledge resources such as ontologies and thesauri (blue). Many resources, such as databases, are designed and populated in a top-down fashion. We could say that a knowledge specialist gave some attention to a set of tasks and a set of potentially-available data when creating

a schema or structure to capture useful items. In contrast, free-text documents are created in natural language. Document authors may have intended to support some particular organizational task but linking the contained knowledge elements to an analyst's query requires a bottom-up approach where documents are organized using manually or automatically generated indexes. Natural-language representations are term-rich but structure-weak as compared to most databases.

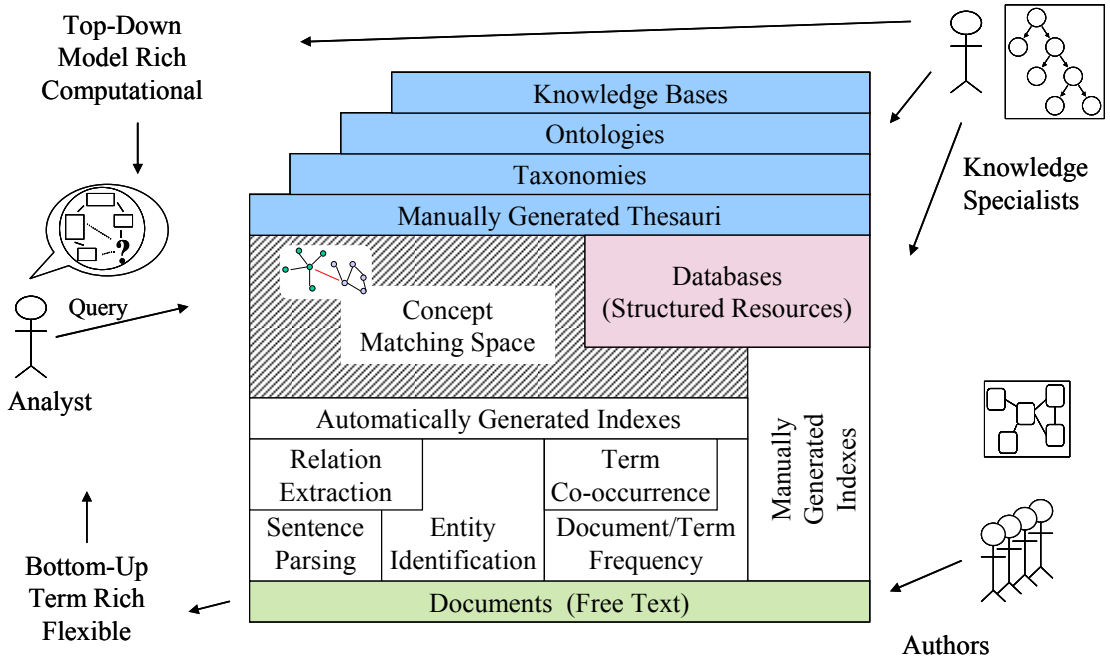


Figure 1.2. The Conceptual Matching Space

Integration requires structure. All of the resources we have discussed can be said to contain models that express domain-appropriate, organizationally-important concepts and relationships. These models contain various levels of structure. Ontologies, taxonomies, and databases are model rich. They gather information into a cohesive

organizational structure or model. The resulting model may or may not match well with other resources or with the goals of new analytic applications. Free text is a term-rich, flexible, model-weak representation. Authors encode data based on their own mental model(s) with varying levels of rigor.

1.3 Informal Node-Link Knowledge Representations

Semantic structures that connect conceptual nodes in relational links (semantic nets) have been used to represent knowledge for hundreds of years. For example, as shown in Figure 1.3, the tree of Porphyry (1239 AD) organizes Aristotle's categories in a visual, node-link picture or model (Sowa, 2000). This general notion of a semantic network can be used to represent a wide variety of information and has been adapted in a significant number of formalisms. We will use the term *concept graph* to refer to a chart in any formalism that consists of conceptual nodes and relation links. We could say that a concept graph "models" the understanding an author has regarding some topic of interest. Creating and using concept graphs has been shown to have positive cognitive impact on users (Chmielewski & Dansereau, 1998; Chmielewski, Dansereau, & Moreland, 1997; Evans & Dansereau, 1991; Ford, Canas, & al., 1995; Hall & Odonnell, 1996). Concept graph structures have been used in education (Ault, 1985; Herl & al., 1999), business (Canas et al., 1998; Lawless, Smee, & O'Shea, 1998), medicine (All & Havens, 1997; Ford, Coffey, Canas, Andrews, & Turner, 1996), law enforcement (Harper & Harris, 1975; Krebs, 2001), and knowledge management (Canas, Leake, & Wilson, 1999; Hoffman, Coffey, Ford, & Carnot, 2001) applications.

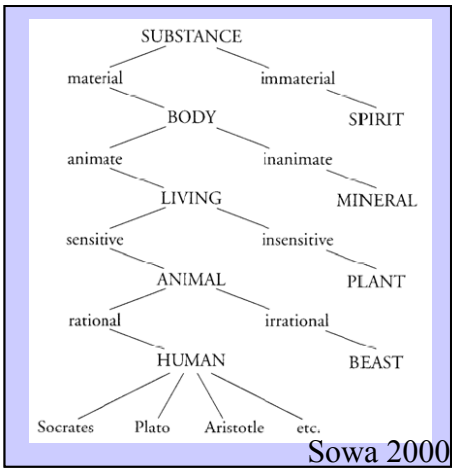


Figure 1.3. Tree of Porphyry, an Early Semantic Net (Sowa, 2000)

1.4 Node-Link Knowledge Formalisms

Node-link knowledge representation formalisms include the existential graphs developed by C. S. Pierce (Hartshorne, Weiss, & Burks, 1931-1935), conceptual graphs presented by John F. Sowa (Sowa, 2000), k-maps (Chmielewski & Dansereau, 1998), W3C's directed labeld graphs (DLG) (Berners-Lee, Connolly, & Swick, 1999), and Novak and Gowin's Concept Maps (Novak & Gowin, 1984). Various formalisms for ontologies and semantic nets also fit in our broad consideration of concept graphs. Expressing and using information encoded in these formalisms requires "mutual understandings between two persons" as highlighted by Pierce's Convention Number 1 (Hartshorne et al., 1931-1935).

Node link knowledge representations have received particular attention in web environments for concept exploration. For example, the WebMap system explored the use of concept graphs in a web browsing interface (Gaines, 1995; B. R. Gaines & M. L.

G. Shaw, 1995; Brian R Gaines & M. L. G. Shaw, 1995) and the Institute for Human Machine Cognition in Pensacola, Florida has studied concept map algorithms for knowledge management (Canas et al., 1998; Canas, Leake, & Maguitman, 2001; Canas et al., 1999; Leake, Maguitman, & Canas, 2002; Leake et al., 2003).

1.5 Informal vs. Formal

Establishing extensive mutual understanding between multiple concept graph creators and users is no easy task, especially when the intent is to support computerized analysis. This is in part because expressiveness trades off with computational complexity in knowledge representation (Brachman, McGuinness, Patel-Schneider, & Borgida, 1999). Rob Kremer's article entitled "Informal vs. Formal" notes a dichotomy between a human user's need to work for a flexible, forgiving, informal system and a computer's need for strong formal semantics (Kremer, 1994). Several kinds of informality are common including:

- ambiguous concept identifiers where different names indicate the same concept,
- imprecise link meanings where the link does not specify all useful details, and
- structural variations where granularity, cardinality, or organizational issues cause semantically equivalent items to be expressed in different graph structures.

Different formalisms allow different levels of flexibility. Model-rich CGs employ strong formalisms to precisely define concepts to support computation. Less formal term-rich CGs allow more flexible expression but as a result conceptual equivalence is not as clearly identified. Still, even when a strong formalism is used, appropriate matching can be difficult as when several nodes represent slight variations of the same basic concept.

In any event, resources which were prepared using different organizational formalisms and assumptions will inevitably suffer from these characteristic informalities.

1.6 Concept Graph Applications

Concept graph representations have been used in a variety of knowledge management applications. Researchers have investigated concept maps in hypermedia systems (B. R. Gaines & M. L. G. Shaw, 1995), concept maps as a part of a knowledge support system (Gaines, 1995), and concept maps for collaboration through map sharing on the web (Brian R Gaines & M. L. G. Shaw, 1995). The use of concept maps for information search and browsing has also been considered in more recent work (Carnot, Dunn, Canas, Gram, & Muldoon, 2001). Concept mapping has been used as part of a knowledge elicitation methodology intended to record, maintain, and exploit expert knowledge (Canas et al., 1999). The CMapTools for concept mapping developed by IHMC (Institute for Human Machine Cognition) have been used in several topical domains including space exploration (http://www.cmex.arc.nasa.gov), medicine (Ford et al., 1996), Naval technician training (Canas et al., 1998), and meteorological investigation (Hoffman et al., 2001). They have been combined with case-based reasoning techniques (Canas et al., 2001) and a map collection has been used to make suggestions as new maps are drawn (Leake et al., 2002; Leake et al., 2003). Another study concludes that "concept mapping should be considered as the interface of choice to a knowledge repository to be used by master's students in Information Management" (Weideman & Kritzinger, 2003). In short, the automatic processing of concept maps for KM continues to be a topic of interest in the KM research community.

The importance of flexible concept graphs for knowledge expression and organization is particularly relevant in educational applications which are firmly grounded in educational theory. A variety of learning theories have been proposed to describe how people acquire knowledge and suggest improved methods for education. Logical positivism focuses on factual knowledge emphasizing observed truths and quantitative data. While this notion of knowledge is a major building block of scientific inquiry, it is less helpful as the basis for knowledge acquisition methods. Educational endeavors are focused on transmitting known information to students rather than adding undiscovered information to an existing body of knowledge. Constructivism focuses on the process by which people acquire knowledge. The constructivist model of learning emphasizes three main ideas (Dalgarno, 2001) which are important for computerized learning systems. First, there is no single "correct" representation of knowledge. Secondly, people learn through active exploration when exploration uncovers inconsistency between experience and current understanding. Finally, learning occurs within a social context. Personal, adaptive knowledge representations require flexible and informal structures.

One important education-related concept graph formalism (or informalism) captures concept maps. Concept mapping was developed as an educational technique to support Ausubel's notion of meaningful learning (Ausubel, 1968; Novak & Gowin, 1984). Concept mapping begins with what the learner knows by calling upon them to extract key ideas and express the key relations between those ideas. An instructor can use what she sees in a student's map as a guide in various instructional activities. Novak and

Gowin note important connections between concept map structure elements such as hierarchy and cross-links and four key meaningful learning processes (Novak, 1998). This close connection between the structure of concept graphs and how people understand things is, perhaps, part of the reason that this kind of representation has been so widely used in educational and other applications.

1.7 Model Matching Techniques

Various methodologies have been proposed to match elements between models. Techniques to accomplish this kind of matching have been explored in the schema or model matching literature. Rahm and Bernstein's review article includes a useful taxonomy in which matching approaches are contrasted based on several classification criteria (Rahm & Bernstein, 2001). This taxonomy is summarized in Table 1.1. Although the first three classification constructs are presented as either/or propositions, a particular matching system may exploit both elements in a hybrid or composite fashion. This taxonomy differentiates various schema matching approaches based on the kind of information used to make the match and characteristics of the resulting matches.

1) A matcher may consider instance or schema information.

2) Element matching or structure matching may be employed.

3) An approach may exploit language or constraints.

4) Matching cardinality differentiates matchers; matches may be 1:1, 1:many, many:1, or many:many.

5) Auxiliary information sources such as dictionaries or user input may be used.

Table 1.1. A Taxonomy of Model Matching Approaches
(Rahm & Bernstein, 2001)

Classification Criteria	Differentiating Characteristic	Example
Instance vs. schema	Use of instance data	Given a column “ssn” in one schema and “empno” in another, instance matching might identify a match between “ssn” and “empno” by noticing that many instances of each column are 9 digit numbers formatted as 111-22-3333
Element vs. structure	Matching granularity	In a database schema match, structure matching might match "Emp with attribute HireDate" (a set of two connected elements) to "Associate with attribute StartedWork" rather than a simpler element match between "Emp" to "Associate"
Language vs. constraint	Element similarity/differentiation algorithms	A constraint function might exclude a match between "AccountNam" and "AccountNm" because one is a string value and the other is a number --a language match would associate AccountNam with AccountNm because their names strings are nearly the same
Matching cardinality	Match cardinality	Accts.AccountNam and Contacts.CompanyName might both be matched to CustomerName in a database schema match if 1:many cardinalities are supported
Auxiliary information	Use of external resources to assist in matching	A domain-appropriate thesaurus might identify "issue" as a possible synonym for "bug" in a software development schema match

1.8 Research Focus

An organization’s ability to effectively leverage available data and information is strongly affected by how those resources are stored and processed. This dissertation focuses on new uses for existing data and information resources. Its guiding focus is the creation of tools that compare, retrieve, and merge existing information in new and useful ways. We emphasize innovative uses of existing resources by developing domain-appropriate analysis algorithms (smart routines using robust data structures) and applying

domain-appropriate visualization techniques. In particular, we will focus on the effective use of informal, node-link knowledge representations. How can existing CG matching approaches be adapted to enhance information integration and analysis applications in light of the informality present in real-world conceptual networks?

This research question is explored in a series of three essays. The first, presented in Chapter 2, proposes and tests a knowledge-anchoring, similarity-flooding algorithm to match instructor and student concept maps. The algorithm has potential to be a key part of a system to improve the efficiency and effectiveness of learning processes that involve student-drawn concept maps. This algorithm is one way to deal with ambiguity arising from node and link name variations as well as organizational variations generated by dozens of students. This essay has implications for the matching of individualized knowledge representations in semantic web and other applications.

The second essay (Chapter 3) presents a framework for organizing biomedical-pathway relations extracted from the abstracts of PubMed journal articles. The usefulness of these relations is hindered by the complex and multi-featured representations used by journal authors as they report key findings. This application addresses this ambiguity using a feature based representation of the extracted relations, large lexicons of biomedical name strings, and a user-controllable aggregation structure. This approach promises to help researchers deal with the vast amounts of available pathway information.

The third essay (Chapter 4) presents our methodology for selecting interesting sub-networks from large collections of criminal associations. Recent events have

highlighted the need for cross-jurisdictional data sharing between law enforcement agencies. We need practical models and methodologies to support investigations in the face of vast quantities of available information. These models can be used to guide the development of useful data sharing protocols and must reflect the real security, privacy, organizational, and scalability issues faced in this important domain. Adding a path-based interestingness evaluation component to previously-studied association closeness measures, the proposed importance flooding algorithm is intended to help crime analysts identify networks of interesting associations using heuristics and a systematic ranking algorithm. This methodology addresses the reality of organizational restrictions on data and the practical need of investigators for understandable, actionable criminal activity network representations.

2 MATCHING KNOWLEDGE ELEMENTS IN CONCEPT MAPS USING A SIMILARITY FLOODING ALGORITHM

2.1 Introduction

In this essay we develop an algorithm for matching knowledge elements in one particular kind of conceptual graph: concept maps created in an educational context. This is an interesting testbed for such an algorithm for at least two reasons. (1) Educational concept maps are minimally formal. Because they are designed to allow maximum expressiveness and application in a wide variety of topical domains, educational concept mapping systems do not enforce naming rules for nodes or links. Therefore concept maps tend to be idiosyncratic in that different people tend create different concept maps of the same topic (Novak, 1998). This is an interesting characteristic because people generally approach query tasks in many of the previously mentioned application domains with idiosyncratic mental models of understanding. (2) Concept maps are relatively easy to obtain. Several computer-based tools have been created to construct concept maps. Improved management tools for collections of concept maps would be potentially useful in a variety of educational applications. Section 2.2 begins by reviewing existing concept map applications and describing our testbed of concept maps. We then examine computational challenges and identify similarity flooding (SF) as a potentially useful map processing algorithm. Our research questions are listed in Section 2.3 and our implementation of SF is described in Section 2.4. Sections 2.5 and 2.6 describe our simulated and student-drawn map experiments, while Sections 2.7 and 2.8 discuss our findings and identify promising future directions.

2.2 Literature Review and Background

2.2.1 Concept Map (CM) Applications

As previously noted, concept mapping was developed as an educational process intended to support a Ausubel's meaningful learning notion (Ausubel, 1968; Novak & Gowin, 1984). The mapping process clarifies a learner's understanding by identifying key concepts and expressing key relations. An instructor can use student maps to focus instructional activities. Four key meaningful learning processes find significant expression in the concept mapping process and in the structural organization of student concept maps: (1) new concept learning, (2) subsumption, (3) progressive differentiation, and (4) integrative reconciliation (Novak, 1998). In concept mapping a learner identifies concepts, hierarchically organizes those concepts, differentiates between them, and expresses more complex cross-hierarchical relations. These cognitive processes are exhibited in multiple layers of hierarchal groupings and in links that connect separate parts of the hierarchical tree.

Empirical studies support the utiltiy of concept mapping as an educational technique. (Chmielewski & Dansereau, 1998) reviews the evidence that concept-map-like representations can be the basis of effective study and learning strategies. Conceptual graphs are effective (1) in cooperative interactions, (2) as pre- and post-study aids, (3) as a substitute for traditional text, and (4) for updating and editing knowledge. Knowledge-mapping training was shown to have positive text processing effects for university students even when maps are not explicitly used. Student concept map scores have also

been compared to standardized test scores. Rye and Rubba surveyed a large number of map scoring methodologies and found some correlation between maps scores and performance on the California Achievement Test (Rye & Rubba, 2002).

Concept maps are used in educational processes where students draw concept maps of course materials and instructors provide feedback to the students based on map content. Evaluating maps is a manual and tedious process (Kinchin, 2001). Computerized concept mapping tools should facilitate computer-supported concept map evaluation. This approach to student evaluation raises an important question: what evaluation measures are both useful and feasible? A wide variety of map scoring techniques have been proposed and evaluated. Shavelson et al. identify not less than 128 possible ways of generating and scoring concept maps (Shavelson, Lang, & Lewin, 1993). Previous research has proposed a variety of map evaluation measures including (1) quantitative measures of map characteristics such as the number of propositions, (2) structural measures such as the number of hierarchical levels found in the expressed relations, (3) correctness measures which reward the validity of a proposition, and (4) similarity to an expert map.

At least one published attempt has been made to evaluate student maps algorithmically (Chen, Lin, & Chang, 2001). The proposed fuzzy integration and fuzzy matching algorithms require an extended concept map formalism referred to as Attributed Concept Maps (ACM). Attributed concept maps are enhanced with importance rankings for each node and link. After carefully constructing a master map from maps created by three experts, fuzzy matching was used to identify how closely each student's map

resembled the master map. Student maps were constructed using node names and relation types from a closed list. Some correlation was found between a student's performance on a handwritten test and the similarity between the student's map and the master map. The analysis suggests that the correlation is more pronounced for high performing students and on more difficult subject matter. The authors note that performance is affected by distinct but correct organizational structures used in different maps; they refer to such variations as "cognitive shifts." It was suggested that the system could be improved by (1) using contextual information for nodes and links, and (2) by incorporating a propositional-based matching mechanism.

Algorithms that find matching knowledge elements concept map pairs could increase the usefulness of concept map collections. Previous CM research has focused on overall map similarity without directly accounting for the differences of vocabulary and representation that commonly occur in human-created knowledge representations. This has been reasonably effective because educational research has frequently employed closed lists of concepts and KM studies generally involve domain experts who tend to share a common vocabulary. We note that the integration of multiple maps requires element matching rather than map similarity calculation. In education, matching elements is important to provide semi-automatic support for the cognitively-oriented map measures by matching the constructs in student constructs and master maps to assess the correctness of drawn links, count levels of hierarchy, identify appropriate layers of progressive differentiation, and recognize cross-links connecting different parts of a generally hierarchical structure. An element-matching algorithm would support

assessment by speeding up the process of assigning a score to a map and support teaching by speeding up instructional feedback.

2.2.2 GetSmart Research TestBed

A collection of data is needed to explore computerized concept map algorithms. Our understanding of the computational difficulties associated with student-drawn concept maps comes in part from our experiences building and using the GetSmart system (Marshall et al., 2003). GetSmart was developed at the University of Arizona as part of NSF's NSDL (National Science Digital Library) project with input from research partners at Virginia Tech. GetSmart supports educational processes by integrating course resources and advanced digital library technologies with concept mapping for personal knowledge representation. More than 100 students at the University of Arizona and Virginia Tech used GetSmart in the fall of 2002. Students created concept maps of course material individually and in groups. Concept maps are envisioned in the GetSmart system (and in this work) as a valuable part of an educational process in which students record their personal understanding and instructors provide feedback based on their expert knowledge. Use of the system in the fall of 2002 resulted in a concept map collection containing 30 or more maps for each of 11 subtopics related to data structures and algorithms, 30 or more maps for each of 10 chapters of an information retrieval textbook, and hundreds of other student-drawn concept maps. Table 2.1 provides usage statistics for GetSmart in the fall of 2002. The time spent evaluating and managing the maps in this large collection points to the real need for improved map processing algorithms.

Table 2.1. GetSmart Usage

114	Student Users
4,000 +	User Sessions
1,400 +	Maps Created for Homework & Class Presentations
50 +	Maps created by groups
40,000 +	concept→link→concept propositions in the maps

2.2.3 Computational Challenges

In this section we will focus on three characteristics of concept maps that make element matching difficult: terminology variation, concept map informality, and organizational variations. Concept maps as implemented in most educational concept mapping systems have labeled nodes and labeled links with no restriction or designation of node types and no restrictions on link names (which are generally not even required). While previous research notes that concept map informality and organizational variations can confound computerized processes, terminology variations seem to have been largely ignored in existing concept map processing routines. As a supplement to previous research we have done some initial analysis of our concept map collection to identify the degree of vocabulary overlap and to compile a list of observed structural variations. This information was useful in guiding algorithm development.

2.2.4 Terminology Variation

People often do not use the same words to represent the same concepts or the same link types. Yet neither the CMapTools approach (Canas et al., 2001) nor the Attributed Concept Map matching routines (Chen et al., 2001) directly address terminology variations. The CMapTools approach sidesteps terminology variation relying

on nearby terms to establish overall similarity for a map pair and the Attributed Concept Maps were constructed with a controlled set of nodes. Because educational concept mapping systems generally do not enforce controlled vocabularies better approaches that deal with terminology variations in a more comprehensive manner are needed. To begin, we evaluated the degree of overlap observed in our collection of maps.

For this evaluation we used work by Furnas et. al as a baseline methodology for assessing the terminology overlap present in the GetSmart map collection. In (Furnas, Landauer, Gomez, & Dumais, 1987) term overlap was measured in several conditions and finding that people choose the same term for the same object about 20% of the time. Because our concept map collection was generated in a classroom setting by students who had all been exposed to the same instructional materials, we expected to see more overlap than was found in the cited study. To measure the overlap in our collection, we chose 4 topical sets of maps and then randomly chose 10 maps from each topical set. We compared the different representations used by different people for the same concept. In some cases the user intended to use the same words for a concept but entered them in the concept map differently. For example the term "E Measure" was entered as (1) "E-Measure," (2) "emeasure," (3) "E Measure," (4) "E Measrue," and (5) "E evaluation measure." All of these representations were considered to represent the same concept. The first 4 were considered to be matching representations while the last was considered to be only somewhat similar. After making some minor spelling adjustments, we found that the most common word for a concept appeared in about 75% of the concept maps. About half of the remaining words were quite similar and the rest were not very similar at

all. Using the measurement methodology described in (Furnas et al., 1987), this equates to between 50 and 60% overlap. This seems to be a reasonable initial estimate of vocabulary overlap for concepts in student-drawn maps.

2.2.5 Informality

Rob Kremer (Kremer, 1994) notes that there is a dichotomy between a human user's need to work with a flexible and forgiving (hence informal) system and the computer's need for a (formal) system with strong semantics. Although concept maps are intuitive and more "computationally efficient" (Lambiotte, Dansereau, Cross, & Reynolds, 1989) than some other forms of presentation such as pure text or predicate logic (Nosek & Roth, 1990), Kremer asserts that concept maps can be computationally enhanced by constraining the "types" of links and nodes that can be created. Leake et al. describe concept map informality by stating that "Concept maps appear similar to semantic nets but have no fixed semantics and vocabulary" (Leake et al., 2002). Concept maps are described by Canas et al. as a "middle point" between structural representations of CBR cases and textual descriptions. "They include structural information and are intended to concisely represent key concept properties but may not use standardized semantics. This makes them more difficult to manipulate autonomously than standardized representations but also easier to acquire when domain experts are called upon to encode knowledge" (Canas et al., 1999). Informality is a problematic but largely unavoidable characteristic of concept maps.

2.2.6 Organizational Variation

Constructivist learning theory asserts that there is no single correct representation of knowledge. This notion corresponds to the "cognitive shift" problem identified for concept maps scoring (Chen et al., 2001). Two people often represent the same concepts using different but equally correct structures. However, examples of organizational variation are not provided in previous literature. In this paper we begin to identify some common organizational variations found in our collection of maps.

We classify variations into four categories: (1) missing elements, (2) added elements, (3) cross-links, and (4) other organizational variations. Without commenting on the correctness of the knowledge expressed in the map snippets, we observe that structural variations are associated with student understanding. Figures 1, 2, and 3 are all based on actual student-drawn concept maps covering tree data structures. Adding and missing elements are depicted in Figure 1. Leaving out important leaf nodes might reflect a student's failure to remember a key concept. Addition of internal nodes can express higher degrees of progressive differentiation. The first map allows for the addition of non-ordered tree traversal methods, the second depicts an additional concept.

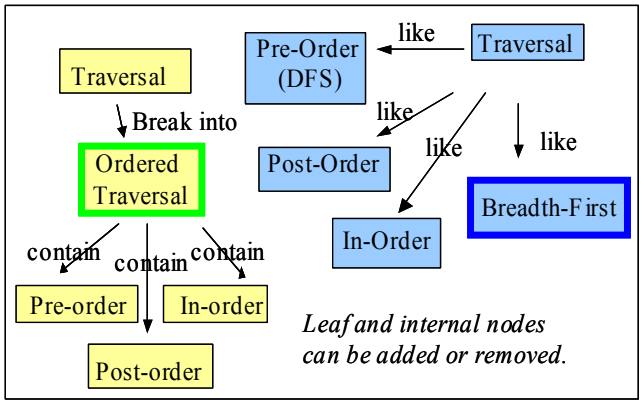


Figure 2.1. Added or Missing Elements
The yellow map has an extra internal node. The blue map has an extra a leaf.

Figure 2.2 shows frequently seen organizational variations. In the first map, three types of binary trees are connected to the higher order concept binary trees in a simple hierarchical structure. In the second, a student uses linear arrangement implying subsumption relationships. The last map reverses the direction of the arrows and shows additional relationships.

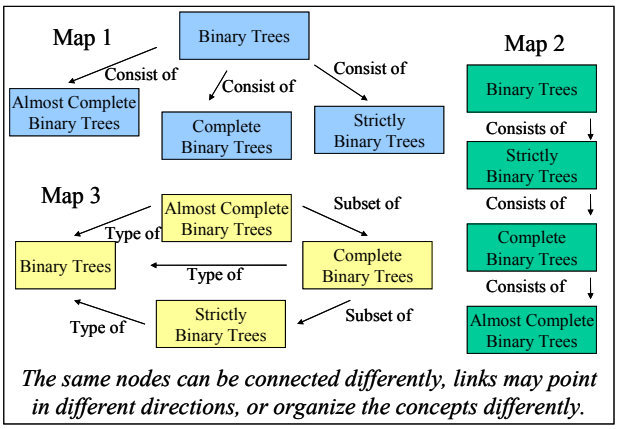


Figure 2.2. Organizational Variations

Figure 2.3 depicts cross-links. In most maps of the topic, students clustered general tree terminology as seen in the children, parent, and sibling nodes. These same maps also usually contain some representation of binary tree types. In Figure 2.3 two hierarchical sections are connected by cross-links. Novak and Gowin (Novak & Gowin, 1984) suggest that this indicates creativity and should be particularly rewarded in concept map scoring.

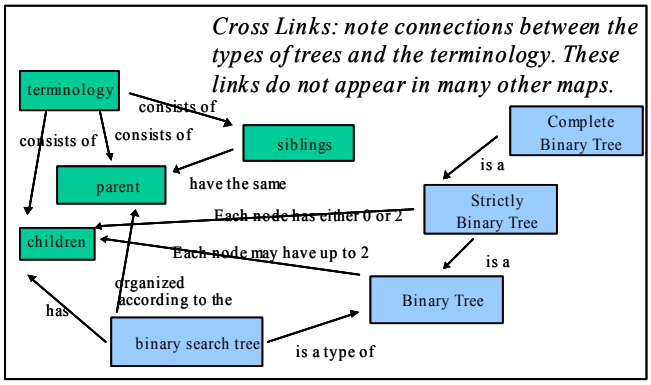


Figure 2.3. Cross Links

2.2.7 Matching Techniques

Having reviewed the challenges associated with element matching in concept maps, we now consider how this matching might be performed. An obvious beginning is comparing node and link labels. Commonly available string matching routines evaluate the "cost" (often character-segment deletions, insertions, and transpositions) associated with converting one string into another. This approach has some obvious limitations. Link names are problematic because a link name is often repeated many times in a concept map and two maps will often label the same relation differently. Logical analysis could also be used for matching graphs. However, even though concept maps can support a variant of predicate logic, most (particularly those generated in an educational setting) lack the formality needed to support logic computations. More promising are schema matching systems that establish mappings between elements in database schemas and conceptual models. They routinely address terminology variation and un-recognized formalism.

Schema matching is a process which creates a mapping between elements of two schemas (Rahm & Bernstein, 2001). In the cited work a schema is defined as a set of elements connected by some structure. That definition clearly applies to concept maps. Schema matching, ontology matching, and representation matching are all used to describe systems or algorithms in this broad area of research. Rahm and Bernstein's work includes a taxonomy covering many existing approaches. Choosing the best matching approach depends on the characteristics of the schemas, the matching environment, and the intended use of the resulting match.

Rahm and Bernstein's taxonomy employs several classification criteria. This taxonomy differentiates various schema matching approaches on based information used and output characteristics. Table 2.2 summarizes the concept mapping implications of the various approaches to schema matching.

Table 2.2. Schema Matching for Concept Map Evaluation

Classification Criteria	Differentiating Characteristic	Concept Map Evaluation Implications
Instance vs. schema	Use of instance data	At the node and link level, each concept map represents its own "schema" with only one available instance therefore schema matching is generally more appropriate than instance matching
Element vs. structure	Matching granularity	Identification of concept→link→concept relations is important because meaningful learning depends on understanding the relationship between concepts and scoring techniques focus on relations
Language vs. constraint	Element similarity / differentiation algorithms	Little or no constraint information is available because educational concept mapping systems generally do not restrict entries in ways which are computationally useful
Matching cardinality	Match cardinality	Because organizational structures of student maps are highly varied, 1:1, 1:m, and m:1 are frequently appropriate match cardinalities for map elements although matches are much more commonly 1:1
Auxiliary information	Use of external resources to assist in matching	Educational concept mapping tools are likely to be used to map knowledge from a wide variety of domains so generic approaches are preferred

Another way of classifying schema matching algorithms is to contrast rule-based vs. learner-based functions. In rule-based systems hand-crafted rules appropriate to a

particular domain or task are employed to implement matching. Two of the many examples of this kind of system are PROM (Doan, Lu, Lee, & Han, 2003) developed by Doan et al, and the PROMPT algorithm implemented as a module of Protégé-2000 (Noy & Musen, 2000). Unfortunately, the informal nature of concept maps drawn as part of a learning process do not provide the kind of detailed validation information needed for rule-based matching. Learning-based methods depend on learnable patterns that persist across different map pairs. It is not at all clear that such patterns are present in student-drawn concept maps.

Based on our review of schema matching systems and an analysis of concept map characteristics, we suggest that an ideal matching system for concept maps would: (1) be schema rather than instance based, (2) allow a matching granularity of at least small map substructures such as concept→link→concept propositions, (3) be language-based because constraints are generally unavailable, (4) would support match cardinalities greater than 1:1 although 1:1 matches would be the norm, and (5) would rely on little or no auxiliary information to maximize generalizability. We propose a similarity flooding algorithm because it meets these requirements.

2.2.8 Similarity Flooding

A new schema matching algorithm called similarity flooding (SF) was proposed in (Melnik, Garcia-Molina, & Rahm, 2002). It matches two directed graphs (schemas, catalogs, or other data structures) to produce a mapping of corresponding nodes. Filters select the best mappings which are then manually reviewed. Algorithm effectiveness was measured in the original SF work by estimating the labor savings obtained using the

algorithm for schema matching tasks. It is an inexact matching approach which relies on the intuition that elements of two graphs are similar when their adjacent elements are similar. Similarity and adjacency are generically defined making the algorithm usable for diverse matching tasks and a good platform for testing new matching heuristics. The effectiveness of SF algorithms for matching structures other than database schemas has not yet been investigated.

The algorithm has four steps: (1) graph representation, (2) calculation of initial similarity, (3) a fixpoint computation, and (4) filtering. To illustrate the algorithm's function, the top of Figure 2.4 shows two small concept maps and the bottom shows how they might be represented during step 1 of the flooding algorithm. Links can also be represented as nodes when preparing the graph for the SF system. This extension would support matching at the concept→link→concept proposition level. Please note that the nodes and node names are abstracted as separate elements. Other available and appropriate attributes can also be attached to the nodes. This allows various attributes to independently contribute to the similarity calculation.

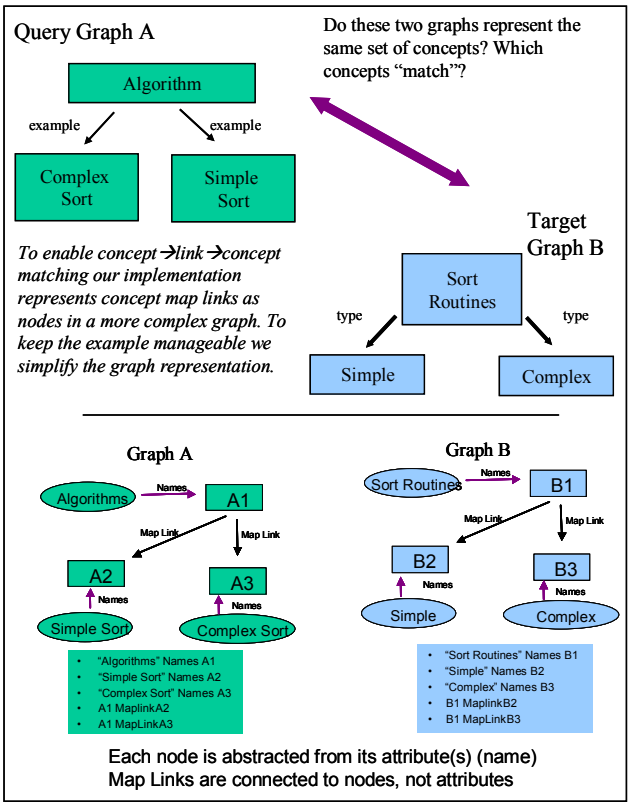


Figure 2.4. Map Representations for Similarity Flooding

Next the algorithm obtains initial similarity values for Graph A/Graph B node pairs. Because our example map includes 6 nodes (as shown in the bottom half of figure 2.4) 36 initial similarity values can be provided. We provided initial similarity only for names (represented as ovals in the figure) and zero values are ignored. Although the initial values strongly affect matching accuracy the other operations are independent of the initial similarity assignment process. Table 2.3 shows the values used in our example. No similarity is provided for the pair *Algorithms/Sort Routines*, and similarity is provided for the incorrect matches *Simple Sort/Sort Routines* and *Complex Sort/Sort Routines*. This emulates the results a string matching routine would provide. One possible enhancement would employ a domain-appropriate thesaurus to connect *Algorithm* and *Routine*.

Table 2.3. Example Initial Similarity Values

Node Pair	Initial Similarity Value Assigned
Simple Sort / Sort Routine	.5
Complex Sort / Sort Routine	.5
Simple Sort / Simple	.7
Complex Sort / Complex	.7

Once the graphs are represented and initial similarity values are established, an iterative fixpoint calculation that updates similarity based on adjacent node similarity creates a mapping between elements. A pairwise connectivity graph is generated, then the algorithm iterates using a fixpoint computation to pass similarity between node pairs until the network stabilizes. A formal description of the algorithm with its internal representation is available (Melnik, Garcia-Molina, & Rahm, 2001). The algorithm operates on the assumption that whenever two elements, one from **Graph 1** and one from **Graph 2**, are found to be similar the similarity score of adjacent elements should be increased. Over a number of iterations, the initial similarity of any two nodes propagates through the graphs. In our example, the similarity initially identified for the node label pairs *Simple Sort/Simple* and *Complex Sort/Complex* propagate to the node pair "A1/B1" to establish the correct final match shown in Table 2.4. Figure 2.5 shows several of the paths along which the similarity travels from names, through node pairs to other node pairs. The output of the computation maps each Graph 1 element to every element in Graph 2.

Because this multimapping is too large for most applications, the fourth and final step chooses which matches to report. Three filters tested by Melnik et. al. resulted in

comparable accuracy: a *Threshold* filter chooses all matches above some threshold value, an *Exact* filter reports the highest match for each node in **Graph 1**, and a *Best* filter requires that each node in **Graph 1** can be matched to only one node in **Graph 2**. The *best* filter uses a greedy algorithm where, for the next unmatched element, a best available candidate is chosen to maximize cumulative similarity. The highest accuracies were reported for the *threshold* and *exact* filters with only slightly lower results reported for the *best* filter. In our example, the highlighted rows in Table 2.4 represent a correct one-to-one mapping and would be chosen by either the exact or best filters.

Table 2.4. Similarity Output – A Multimapping

Graph A Node	Graph B Node	Similarity Output
A1 (Algorithms)	**B1 (Sort Routines)**	**1.00**
A1	B2 (Simple)	0.10
A1	B3 (Complex)	0.10
A3 (Complex Sort)	**B3 (Complex)**	**0.65**
A3	B2 (Simple)	0.40
A3	B1 (Sort Routines)	0.07
A2 (Simple Sort)	**B2 (Simple)**	**0.40**
A2	B3 (Complex)	0.43
A2	B1 (Sort Routines)	0.15

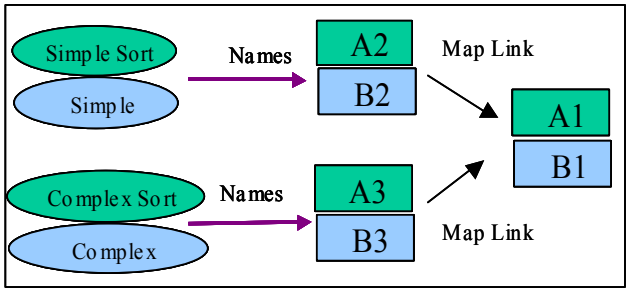


Figure 2.5. Selected Paths in the Similarity Propagation Graph

Melnik et. al evaluate four variations of the fixpoint calculation. *Basic*, *A*, *B*, and *C* are shown in Table 2.5. The function *f* increments the similarity of an element pair (σ^{i+1}) based on the similarity of its neighbors. The relative influence of the initial similarity value (σ^0) and the previous iteration's value (σ^i) changes in each variation. *C* is most strongly influenced by the initial similarity values. *Basic* was found to be the slowest to converge and the least accurate. *A*, *B*, and *C* had comparable convergence properties but *C* was slightly more accurate.

Table 2.5. Fixpoint Formulas

Identifier	Fixpoint Formula
Basic	$\sigma^{i+1} = \text{normalize}(\sigma^i + f(\sigma^i))$
A	$\sigma^{i+1} = \text{normalize}(\sigma^0 + f(\sigma^i))$
B	$\sigma^{i+1} = \text{normalize}(f(\sigma^0 + \sigma^i))$
C	$\sigma^{i+1} = \text{normalize}(\sigma^0 + \sigma^i + f(\sigma^0 + \sigma^i))$

2.3 Research Questions

Our review of the literature and our experiences with concept mapping in education suggest that the similarity flooding algorithm can be used to support concept map management processes by matching knowledge elements found in concept map pairs. We explore this potential by experimentally addressing three research questions:

1. Given query and target concept maps, can we correctly identify node and link knowledge elements from the query map in the target map?
2. How does the similarity flooding algorithm perform for this matching task?

3. How do commonly-observed concept map organizational variations affect the accuracy of the matching process?

We explored these questions in two experiments. The first employed simulated concept maps to test algorithm performance in a controlled setting. The second evaluated the algorithm using student-drawn concept maps.

2.4 Implementation

In the graph representation phase each concept→link→concept proposition found in the map was presented to the algorithm three ways. First a node name and maplink relations are created (as shown in Figure 2.4). Then separate link elements are created and connected to the nodes to represent each proposition.

In initial tests we tended to match to "superstructures." That is, viewing the graph as a somewhat hierarchical structure, we occasionally had incorrect matches to items at "higher" levels in the map structure. This is a documented tendency in the SF algorithm (Melnik et al., 2002). We addressed this problem by generating "hierarchical structure" elements when one node was connected to three or more nodes of the same color by links with the same name and direction, as shown in Figure 2.6. Existing concept maps, as drawn by students, require no special input or manual adjustments to identify these structures. We also introduced a "node anchoring" mechanism. Key terms and commonly used abbreviations are identified as anchor points. Whenever these terms are found in both query and target maps, they are "locked-in" as best matches. We increase the match value for these pairs in each iteration of the fixpoint computation.

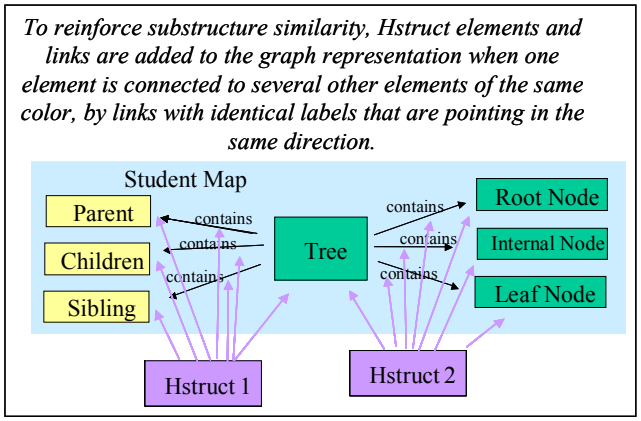


Figure 2.6. Reinforcing Substructures

2.5 Simulation Experiment

Our simulated map experiment evaluates SF's capabilities for matching elements in concept maps under different structural and terminological variations. First we verify functionality with concept maps because the algorithm was previously used on data schemas. We wanted to see if the algorithm increased concept map element matching accuracy above the initial similarity evaluation. Next we evaluated filtering methods and fixpoint formulas. Finally we evaluated performance in the presence of organizational structure variations. For this experiment we created a set of systematically altered maps and 30 sets of randomized initial similarity values.

2.5.1 Simulation Experimental Design

The simulated map set was designed to be representative of a collection of human-drawn concept maps. In Section 2.2.3 above we identified three crucial concept map characteristics that have implications for computational performance: informality, structural variation, and terminological variation. To represent a concept map collection

we began with the prototype map shown in Figure 2.7. Actual student maps only occasionally show such a clear hierarchical structure. However, our review of concept map scoring techniques and concept mapping benefits suggests that hierarchical arrangement is the most important map characteristic. The prototype is intended to represent an instructor-created expert or master map. Its nodes are named generically so it can represent a map on any topic. One of the key notions of the simulation is separation of the effects of map structure variation from the effects of initial node similarity values. By naming the nodes generically and assigning randomized initial similarity values we allowed the comparison of these effects. The node labeled "Root" might represent the top-level concept for a map on any topic. The cluster of yellow nodes (YellowRoot, Y1, Y2, and Y3) stands for some grouping of important concepts related to the "Root" concept. The rest of the maps in the simulation are intended to mimic commonly observed, cognitively important organizational variations.

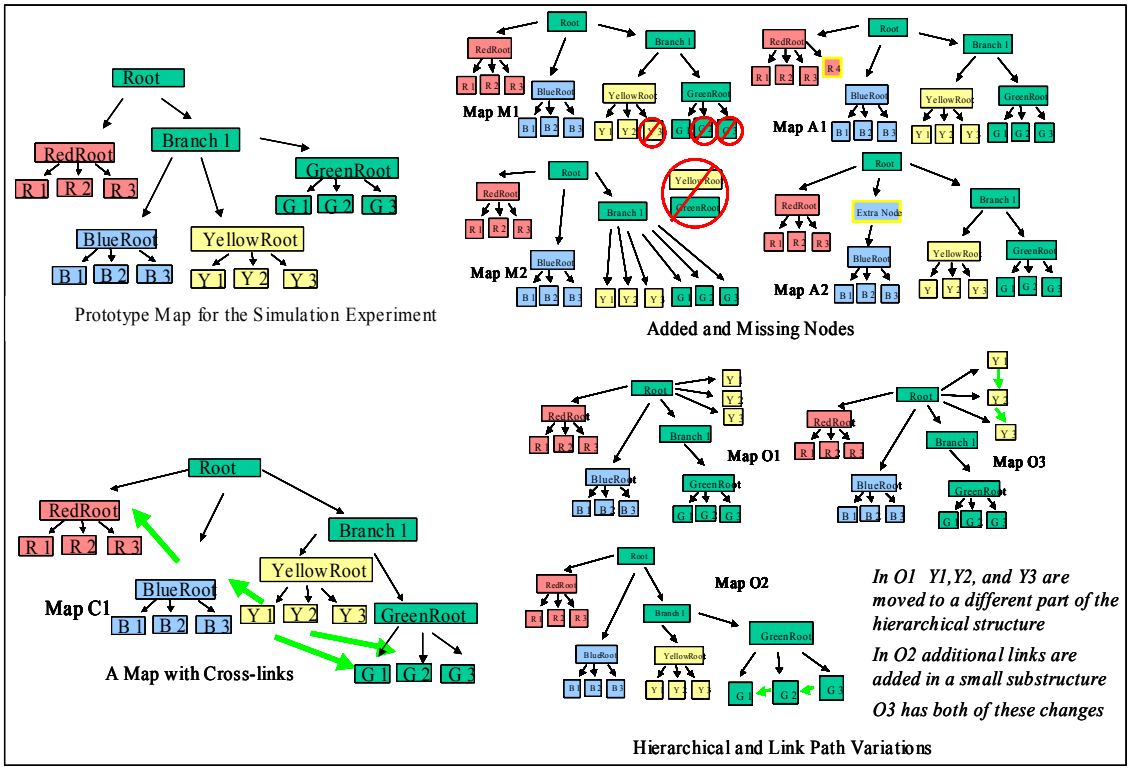


Figure 2.7. Maps Used in the Simulation

We compared the prototype map to 9 simulated maps. In maps A1, A2, M1, and M2 we added or removed nodes. Map C1 includes cross-links connecting different portions of the hierarchy. In O1 we attached the Y1, Y2, and Y3 nodes to the root node. In O2 we added additional links between nodes within a cluster and in O3 we made both of these changes together. Finally, we used an exact copy of the prototype map.

The second part of our simulated environment is a set of initial similarity values designed to simulate string match results from a set of concept maps drawn by different people on the same topic. Each comparison requires a matrix relating each node in the query map to each node in the target map. The matrix contains initial similarity values for nodes representing the same concept and values for nodes representing different

concepts. For example, consider the node labeled *GreenRoot* in the simulated map. The simulated map M1 also has a node labeled *GreenRoot*. If the prototype map were an instructor's master map (query map Q) and map M1 was a student's map (target map T), the student might have used the same term exactly, a very similar term, or a completely different term to represent *GreenRoot*. A correct result matches *GreenRoot* from the query map to the *GreenRoot* node in the target map.

To calculate values for same-concept node pairs (ie. GreenRoot in some map Q to GreenRoot in some map T) we used the estimates described in Section 2.3.1. Table 2.6 describes the distribution of these values. An initial similarity value of 1 was assigned to 75% of the same concept nodes to represent those occasions when the same term is used to denote the same concept. A value of .85 was assigned to 15% to represent very similar representations. The rest were assigned .173 to simulate cases where different terms have been used. This distribution approximates the 50-60% vocabulary overlap found for same-concept nodes in a set of student-drawn maps. Table 2.7 describes the initial similarity value distribution used for nodes representing different concepts (ie. GreenRoot in some map Q to R1 in some map T). We created this distribution using string match calculations performed on more than 25,000 node label pairs found in a set of 60 topically-similar concept maps. Figure 2.8 shows a portion of the initial similarity value matrix for one of the 30 computations for altered map A1.

Table 2.6. Simulated Initial Similarity for Same Concept Node Pairs

Simulated Map Nodes		Examples of What the Simulation Represents	Possible Initial Similarity Value
Node Pair			
Prototype Map Node	Target Map Node		
R1	R1	"Pre-Order", "pre-order"	1
R1	R1	"Pre-Order", "pre-order (DFS)"	0.85
RedRoot	RedRoot	"Traversal Method", "Algorithm"	0.175

Simulated matching nodes are assigned initial similarity based on analysis of the vocabulary overlap found in 40 maps covering 4 topics (equates to 50-60% overlap)

75% - Match Value 1.0 Nearly Exact Terminology
15% - Match Value .85 Very Similar Terminology
10% - Match Value .175 Significantly Different Terminology

Table 2.7. Simulated Initial Similarity for Different Concept Pairs

Simulated Map Nodes		Examples of What the Simulation Represents	Possible Initial Similarity Value
Node Pair			
Prototype Map Node	Target Map Node		
R1	R2	"Pre-Order", "post-order"	0.85
R1	Y3	"Pre-Order", "parent"	0.51
Root	G3	"Tree", "Sibling"	0

Simulated non-matching nodes are assigned initial similarity based on an analysis of string match values from 25,000+ node label pairs found in a set of 60 topically-similar concept maps

One Set of Initial Similarity Values
Altered Map A1

Prototype Map

Node Identifiers	Root	RedRoot	R1	R2	R3	R4
Root	1	0.057	0.057	0.12	0.245	0.12
RedRoot		0.85	0.12	0.12	0.12	0.307
R1			1	0.12	0.057	012
R2				0.175	0.495	0.12
R3					1	0.245
…						

The shaded values are randomly drawn from the same-concpet distribution (Table 2.6), the other values are randomly drawn from the different-concept distribution (Table 2.7). A "perfect" match would have 1 in the shaded boxes and 0 in the others. In practice, the same-concept node pairs only tend to have higher values than different-concept pairs.

Figure 2.8. An Initial Similarity Matrices for Altered Map A1

Each of the simulated maps was tested with 30 different sets of randomized initial similarity values. That is, each generated target map was compared to the prototype map 30 times. In the second step of the similarity algorithm, initial similarity for same-concept node pairs were assigned values from the first distribution, while values from the second distribution were used for different-concept pairs.

2.5.2 Simulation Experiment Results

2.5.2.1 Verification of Algorithm Functionality

The similarity flooding (SF) algorithm performed as expected for the different fixpoint formula options. Formulas *A*, *B*, and *C* showed comparable recall accuracy, substantially out-performing the *Basic* formula as shown in Figure 2.9. Accuracy was measured by dividing the number of correctly matched nodes by the number of correct matches possible. Figure 2.10 shows that, as in the previous SF study, the algorithm converged more quickly (in fewer iterations) when employing Formula C.

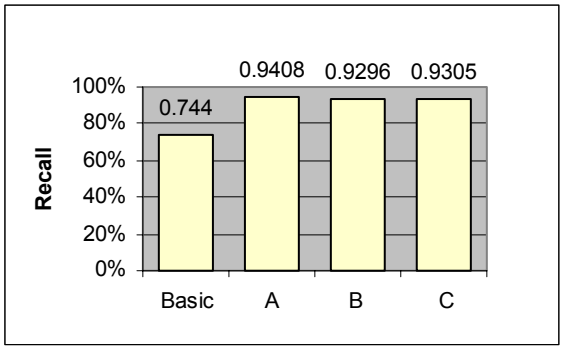


Figure 2.9. Fixpoint Formula Accuracy

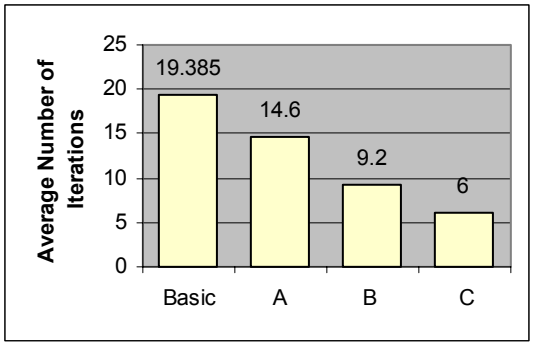


Figure 2.10. Fixpoint Formula Convergence

The *Best* and *Exact* filters were compared because they both produce at most one match in the target graph for each node in the query graph while the *Threshold* filter allows multiple results for each query node. Employing the *Best* filter improved matching accuracy over *Exact* filtering. *Best* improved 132 of 240 mappings (55%) of those matches while reducing the accuracy of only 24 (10%). *Best* produced a net increase of 210 correctly matched nodes on 4,650 attempts (4.5%). Table 8 shows the improvement in node match recall for the *best* filter as compared to the *exact* filter. All the improvements were significant at the p= .05 level.

Table 2.8. Filter Recall Results

	Node Recall		
	Best	*Exact*	Improvement
Nodes	.92	.88	4 %
Links	.88	.78	10 %
Elements	.93	.83	10%

The similarity flooding algorithm improved on the node matching accuracy over a match based only on the initial similarity values as shown in Table2.9. To create a comparable result, we used the *Best* filter for both the similarity flooding and string match results. Improvement ranged from 3 to 11 percent; the improvement in the average

accuracy for the 30 trials of each of the 9 maps was found to be significant at the p=.05 level.

Table 2.9. Correct Node Match Ratio, Similarity Flooding vs. Initial Similarity

Map Variation	SF Result	Initial Value	Improvement
Identical Graph	0.95	0.84	11%
A1 Added Leaf Node	0.94	0.85	9%
A2 Added Internal Node	0.91	0.84	7%
M1 Missing Leaf Nodes	0.94	0.86	8%
M2 Missing Internal Nodes	0.86	0.83	3%
C1 Cross-links	0.95	0.86	9%
O1 Moved Node Group	0.91	0.83	8%
O2 Added Links Within a Substructure	0.96	0.89	7%
O3 Two Organizational Variations	0.93	0.84	9%

2.6 Student-Drawn Map Experiment

The student map experiment tested the algorithm in a more complex and realistic environment. We wanted to see if a similarity flooding match was better than a match based on actual string match values. We also wanted to identify situations in which the algorithm returned inaccurate matches to identify strategies for improvement.

2.6.1 Student-Drawn Map Experimentation

Thirty topically similar, student-drawn target maps from the GetSmart collection were selected. They exhibited a variety of terminological and structural variations. We created a query map of the topic emphasizing hierarchical elements frequently found in the student maps. Hierarchical relations were emphasized because of their cognitive and educational importance. Two graduate students familiar with the topic compared the query map to each of the target maps. A list of correct matches was compiled; only matches agreed upon by both reviewers were used in calculating accuracy results. The query map was matched to each of the student maps using our similarity flooding

implementation. The *Best* filter was used in both similarity flooding (SF) and string match (SM) processing. A commonly available string match algorithm provided similarity scores for each node in the query map to every node in each of the target maps. In addition, we compared the full text of each concept→link→concept proposition pair. For example the node label pair (*Traversal*, *Ordered Traversal)* was assigned an initial similarity of .529 by the string matching algorithm, and the proposition pair (*Ordered Traversal contains In-Order*, *Traversal include In-Order)* was assigned .559.

2.6.2 Student-Drawn Map Results

The similarity flooding (SF) based matching system out-performed the string matching (SM) algorithm for both concept nodes and concept→link→concept propositions. SF & SM were somewhat complementary. SM occasionally identified correct matches missed by the SF algorithm. Table 2.10 compares the SM and SF recall. SF+SM reflects occasions when either the SM or the SF result was correct. All of the SF results are significantly better than the SM results at the p= .05 level.

Table 2.10. Recall Score of SF vs. SM for 30 Student-Drawn Concept Maps

	SF	SM	Improvement	SF+SM
Nodes	.94	.88	6%	.95
Propositions	.79	.50	29%	.84
Combined	.88	.72	16%	.91

Recall is the more important measure of accuracy because it would be relatively easy for an instructor to ignore incorrect matches when evaluating a map. However, recall can also be measured for various levels of precision. A minimum similarity threshold would cause the algorithm to ignore many poor matches. Recall at various

precision levels is reported in Figure 2.11. By selecting an output similarity threshold of .8 we achieved both recall and precision above .90.

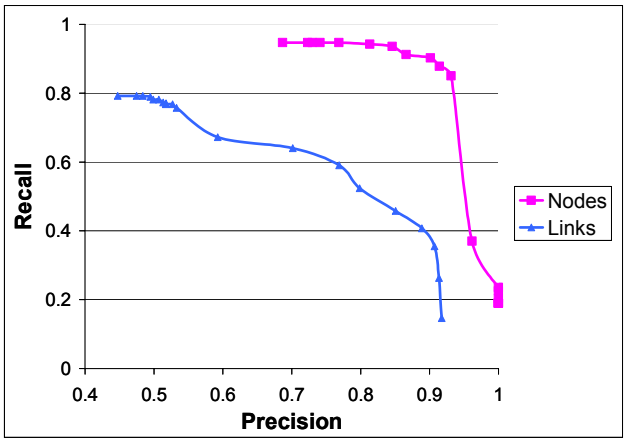


Figure 2.11. Recall vs. Precision

2.7 Discussion

Although the matching results are encouraging, they can still be improved. We reviewed the remaining matching and identified four recurring problems. Incorrect matches can be traced to (1) student misconceptions (2) synonymy, (3) cardinality, and (3) granularity. Because informality and flexibility enhance the educational value of concept mapping, our observations are intended to identify methods of increasing match accuracy without imposing restrictions on the student map building process.

Student misconceptions are exhibited in incorrect links and lack of organizational clarity. Some matching errors could be directly traced to factually incorrect links that introduced noise in the algorithm. Removing incorrect links (for example a link identifying a BTree as a type of Binary Tree) would have increased node matching accuracy in some cases. Map clarity is also important. Flat trees with few hierarchical

levels reflect a lack of conceptual differentiation. Identifying such ambiguous or incorrect representations would be helpful educationally and increase matching accuracy.

Different people frequently use different abbreviations, synonyms, or word forms in node labels. Examples include *CBT* for *Complete Binary Tree*, *Routines* for *Algorithms*, and *Trees* vs. *Tree*. In many cases the system corrects these errors, but not always. For example, the term pairs *child/descendant* and *parent/ancestor* are presented in lecture as contextually equivalent terms for a concept. The string match algorithm matched *child/ancestor* over *child/descendant*, and *parent/descendant* over *parent/ancestor*. Because the concepts were placed in equivalent structural positions, the SF algorithm could not correct the match. Even when a correct match is found for such a node pair, ambiguous signals may be introduced into the similarity propagation graph causing other errors. A query map could be structured to include domain appropriate synonyms to help with these easily-anticipated problems and lexical resources could be used. We observed that locking in matches for key terms (i.e. “tree” always matches with “trees” and “binary tree” always matches with “binary tree”) increased overall matching accuracy. Introducing just these two key terms for enhanced matching resulted in a substantial increase in overall matching performance in one set of maps. Key top-level terms could be listed in the query map or inferred from a map collection prior to processing the individual map pairs.

Matching cardinality also affects matching accuracy. We found that *Best* filtering (which enforces a 1:1 match cardinality) improved accuracy. But, our initial review of appropriate characteristics for a matching algorithm for concept maps (presented in

Section 2.2.5) noted that 1:M and M:1 would be appropriate on some occasions. Out of 30 maps on tree structures from the GetSmart collection, 3 maps included *Btrees* and *B+Trees* in the same node *B/B+Trees*. The fixpoint calculation portion of the SF algorithm gave a high similarity score to both "Btree" → "B/B+Tree" and to "B+Tree"→ *B/B+Tree* but the *Best* filter forced it to choose only one of the matches. In addition to the main error (missing the match between *Btree* and *B/B+Tree*), the resulting ambiguity occasionally caused other mismatches.

In this initial implementation we restricted granularity by allowing matches only between nodes and nodes or propositions and propositions. In a number of cases, 3-element knowledge structures (node→link→node propositions) would have been better matched to 5 element structures (node→link→node→link→node). For example the query map includes the proposition: (*tree → include → binary tree*). In several maps an additional node has been inserted (*tree → has → types → includes → binary tree*). These intermediate nodes frequently labeled words like *types*, *examples*, *terminology*, or *comparison*. Rather than introducing new concepts, these nodes clarify the relationships between other concepts. It may be useful to compile a list of these words and use that list to automatically adjust the map representation provided to the flooding algorithm.

2.8 Conclusions and Future Directions

Structural matching with the SF algorithm presented in this work is a promising approach for matching concept map elements. The system improved on simple string match results but employed only readily available information such as common abbreviations, key terms, and node colors.

This work identifies a need for element level matching in concept maps and it explores the use of schema matching techniques. Existing concept map evaluation techniques were reviewed for common themes and measures. Existing schema matching algorithms were reviewed in a concept map matching context. A candidate algorithm, the similarity flooding algorithm developed by Sergy Melnik, Hector Garcia-Molina, and Erhard Rahm, was identified and tested to establish a performance baseline for future work.

Previous computerized concept map applications measure overall map similarity but do not emphasize the element matches needed in providing student feedback. Previous work has generally considered only conceptual nodes and not propositional links. Our proposed approach identifies these links as a computationally important dimension of the knowledge contained in a concept map. Also, in contrast to previous concept map algorithms, our system uses concept maps as they are generally created in educational settings without controlling the list of potential nodes or adding importance weights to the maps. Even so, in our experimentation with student-drawn maps, we were able to identify 91% of the correct node and proposition matches.

To guide improvements to the matching process, we identified a few commonly occurring organizational variations and noted their relationship to educational and cognitive processes. We tested both simulated and human drawn concept maps exhibiting these variations. Because these variations negatively impacted matching accuracy, the system was adjusted in several context-appropriate ways to increase accuracy. We introduced “node anchoring” to lock in key terms and automatically recognized some

important hierarchical clusters using node colors, link names, and link directions. Using a matching algorithm such as the one described in this work, educational map evaluation and feedback processes might be improved. Mapping suggestions might also be provided for students by leveraging a collection of existing maps. It is hoped that this kind of prompting or tutoring will have a positive effect on learning and knowledge acquisition.

We plan to implement element matching in a semi-automatic scoring system and measure its impact on student feedback processes. Element matching is needed for such a system. Establishing a mapping between single nodes and between concept→link→concept substructures is a good beginning for matching larger structures such as hierarchical clusters. We plan to augment our mapping system to leverage the SF multimapping to identify some of these larger substructure matches. Although our current implementation individually matches a query map to a target map, information gathered in matching one map may be useful in matching other maps in the same collection. We plan to explore this possibility in the system as it is developed. Finally, because student misconceptions are educationally important and have a negative effect on matching accuracy, we intend to add misconception detection capabilities to the concept map evaluation system.

2.9 Acknowledgements

We would like to thank the NSF for supporting this project. NSF National STEM Education Digital Library: “Intelligent Collection Services for and about Educators and Students: Logging, Spidering, Analysis and Visualization” Award No. DUE-0121741,

Program 7444. September, 2001-August 2003. We also would like to thank the GetSmart team and other members of the U of A's AI Lab who built GetSmart components, especially Benjamin Smith, Chun Q. Yin, and Steven Trush. Finally, we recognize and appreciate the efforts of Ed Fox, Rao Shen, and Lillian Cassel in evaluating the GetSmart system and providing important feedback guiding its development.

3 AGGREGATING AUTOMATICALLY EXTRACTED REGULATORY PATHWAY RELATIONS

3.1 Introduction

The number of new abstracts appearing each day in the PubMed database rose from an average of 746/day in 1980 to 1,760/day in early 2005. To help researchers leverage this vast and growing collection of documents, several systems have been developed to extract biological relations from free text (see Section II.A). These systems promise decreased costs and increased coverage as compared to the manual curation processes. Considerable attention has been paid to accuracy of these systems considering both the correctness of extracted information (precision) and the coverage of the output (recall). The evaluators ask "is it correct?" and "did we get everything?" As these extraction technologies mature researchers need to go beyond accuracy evaluation and consider system usefulness.

The GeneScene system extracts regulatory pathway triples from MEDLINE abstracts to support search, visualization, knowledge discovery, and automatic analysis algorithms so that researchers can more efficiently leverage available information to gain insight from previous work, generate new hypotheses, and analyze experimental results. To our knowledge, it is the only end-to-end system that automatically extracts pathway relations from abstracts and presents them as a network. We are scaling up our system to handle millions of abstracts. GeneScene users suggested that extracted relations would be more useful in accomplishing these tasks if (1) references to the same substances and

functions are indexed appropriately, (2) those references can be directly connected to existing database resources, and (3) important contextual information is included.

In this paper we propose a methodology for meaningfully organizing or "aggregating" the relational output of biomedical relation extraction systems and present an initial evaluation of the BioAggregate tagger which decomposes the relations to identify features to support the organization process. Section II describes the output formats of several current systems and relevant lessons learned in biomedical object recognition research. Later sections list research questions, outline the functionality of our aggregation system, describe the testbed used for evaluation, and provide some preliminary evaluation of the effectiveness of our approach.

3.2 Background

Building a complete system for the processing of free text into a useful molecular pathway network is a multifaceted task. (Rzhetsky et al., 2004) summarizes many of the related issues in outlining the architecture of the GeneWays system. Two of the key processes are relation extraction and biomedical object recognition.

3.3 Relation Extraction Output

The systems listed in Table 3.1 use various Natural Language Processing (NLP) techniques to extract the relational information from free text (Friedman, Kra, Yu, Krauthammer, & Rzhetsky, 2001; Gaizauskas, Demetriou, Artymiuk, & Willett, 2003; Leroy, Chen, & Martinez, 2003; McDonald, Chen, Su, & Marshall, 2004; Palakal, Stephens, Mukhopadhyay, Raje, & Rhodes, 2003; Park, Kim, & Kim, 2001; Pustejovsky,

Castano, Zhang, Kotecki, & Cochran, 2002; Rindflesch, Tanabe, Weinstein, & Hunter, 2000). The systems at the top of the table extract triples containing two named entities and a labeled connector that describes the relationship. This format is frequently used in the visualization and automatic analysis of biomedical information (Lanckriet, Deng, Cristianini, Jordan, & Noble, 2004). Although the systems listed in the bottom half of the table create nested predicates, the GeneWays developers acknowledged that the predicates need to be unwound into binary relations (i.e. relational triples) before they can be organized into a network (Friedman et al., 2001).

Table 3.1. Relation Extraction Systems

Relation extractions systems generally produce either relational triples or complex predicate relations, complex predicates are "unwound" for aggregation

	System	Method	Output
Relational Triples	Medstract: Pustejovsky et al.	Semantic Automata	• Relational Triples for Inhibition Relations
	Palakal et al.	POS Tags & HMM Co-reference Grouping	• Verb-labeled Relational Triples
	GeneScene: Leroy et al.	Sentence Parsing, FSA Emphasizes closed class words	• Relational Triples With Negation
	Arizona Relation Parser (ARP): McDonald et al.	Hybrid Syntax/Semantic Parsing	• Relational Triples With Negation, name strings frequently include several modifiers
—	GENIES: Friedman et al.	Semantic Extraction Templates	• Complex Predicate Relations are "unwound" by GeneWays into labeled binary statements
Predicates	Park & Kim	Combinatory Categorical Grammar (CCG)	• Predicate Relations e.g., Activates(A,B), or Activates(A,B,C)
	Edgar: Rindflesch et al.	Matching to UMLS	• Appears to be predicate relations
	PASTA: Gaizauauskas et al	Semantic Templates	• Emphasizes "feature" relations e.g., "mutated-p53"

3.3.1 Biomedical Object Recognition

Effectively organizing relations depends on correctly matching entities and connectors. In related work, systems that recognize or identify biomedical name strings in

text have been the subject of significant research efforts. Entity "recognition" systems find bits of text referring to biomedical objects and "identification" systems associate name strings with known biomedical objects (Tuason, Chen, Liu, Blake, & Friedman, 2004). While recognition tasks can be accomplished with approximately 80% accuracy (Yeh, Hirschman, Morris, & Colosimo, (2004)), (Hirschman, Morgan, & Yeh, 2002) reports only 2% - 29% accuracy in matching fly gene and protein name strings to items in a corresponding lexicon of substance names.

Biomedical named entity recognition systems face three key problems (Palakal et al., 2003): (1) new or unknown words, (2) compound word recognition, and (3) ambiguous expressions. Biomedical name strings are frequently composed of several terms (Ogren, Cohen, Acquaah-Mensah, Eberlein, & Hunter, 2004). To address these problems biomedical information extraction systems employ extensive lexicons and leverage character patterns and frequently occurring words. For example, the PROPER system (Fukuda, Tsunoda, Tamura, & Takagi, 1998) employs a list of f-terms (e.g. gene and protein) and character patterns (e.g. a numeric digit following 3 alphabetic characters) to identify the word boundaries of phrases that refer proteins.

Nearly all biomedical information extraction systems use lexical resources to identify biological object references. While some available resources (such as the Gene Ontology) implicitly or explicitly identify object classes such genes and gene products, other resources enumerate instances of those classes. For example, LocusLink lists genes and RefSeq lists genes and gene products. These lists are subject to term ambiguity where multiple substances share a common name string. Ambiguity is even more pronounced

across several lexicons (Hanisch, Fluck, Mevissen, & Zimmer, 2003) with reported cross-dataset ambiguity between 4-20% and overlap with common English words from 0% to 2.4% (Tuason et al., 2004). This kind of ambiguity is more pronounced among those terms that are both included in the lexicons and used in MEDLINE abstracts (Marshall, Su, McDonald, & Chen, 2005).

3.4 Research Questions

Our review of existing biomedical information extraction systems leads us to conclude that several factors negatively impact the usefulness of the extracted relations: (1) biomedical object and relational connector name strings are represented by various synonyms and contain potentially confusing modifiers; (2) some relations involving the same entity pairs seem to conflict with each other, especially when contextual information is ignored; (3) it is difficult to link extracted relations to other data sources (e.g., a genome, publication, or pathway databases).

This study employs a systematic approach to relation aggregation to find out how effectively we can aggregate automatically-extracted biomedical relational triples. We want to:

- Index multiple references to the same object over expected variations in relational granularity.
- Connect the relations to existing ontological resources.
- Capture contextual information.

An effective system would reduce the number of items needed to display extracted information, highlight relative importance based on frequency of occurrence in the biomedical texts, and format the relations for use in knowledge discovery algorithms.

3.5 System Design

The BioAggregate tagger implements a feature decomposition approach to biomedical concept matching as part of the larger GeneScene system depicted in Figure 3.1. The Arizona Relation Parser (ARP) (McDonald et al., 2004) extracts relations, the BioAggregate tagger annotates those relations with feature assignments to support aggregation, and the visualizer allows users to view the extracted and organized results.

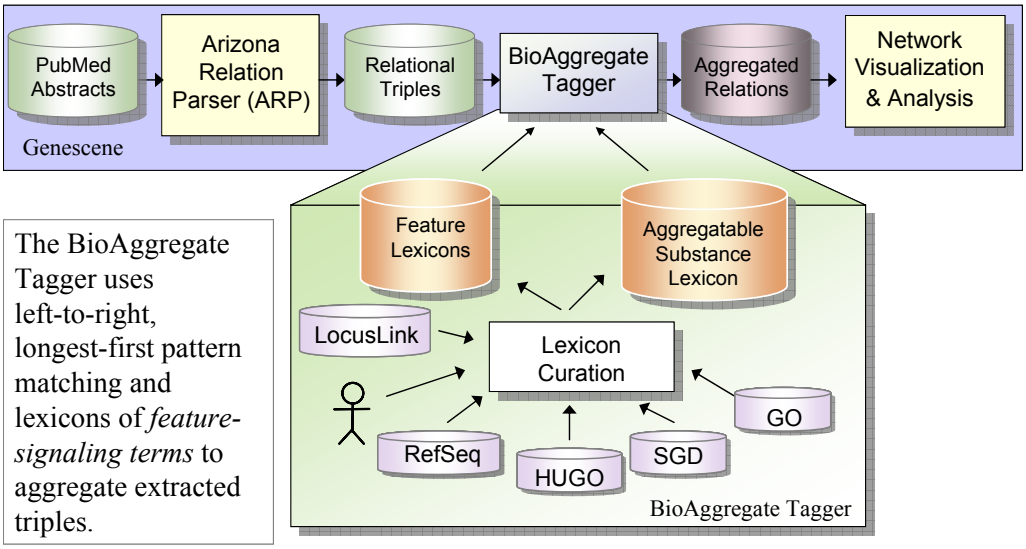


Figure 3.1. The GeneScene System
Genescene supports extraction, organization, and visualization of pathway relations found in the text of MEDLINE abstracts. The BioAggregate tagger organizes the relations to improve research utility.

The BioAggregate tagger decomposes a relation's entity and connector name strings by recognizing words and phrases that signal features. We will refer to such terms as feature-signaling terms. The tagger implements three noval notions: aggregatable substances, pseudo-substances, and residuals. These notions are closely related to the key entity recognition challenges identified in previous research. The key components of the tagger are lexicons and an efficient finite state automata (FSA) algorithm as depicted in Figure 3.2. Note how both entities and connectors in the relation are decomposed into a set of features.

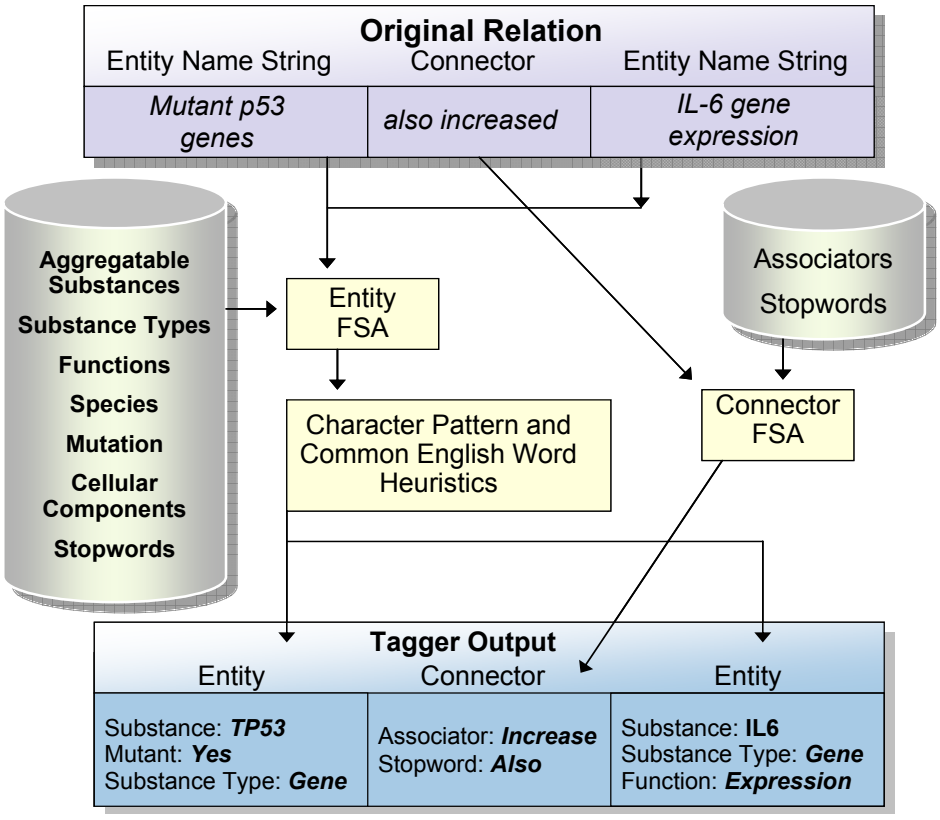


Figure 3.2. The BioAggregate Decompositional Tagger
We identify a relation's features by matches its words to terms in extensive feature term lexicons using an efficient Finite State Automata (FSA) algorithm.

3.5.1 Feature Lexicons

Extensive feature lexicons drive the aggregation tagging process. We manually reviewed a large number of extracted relations while developing a list of desired features. A number of possible features were considered; we preferred features that (1) frequently occur in the words near substance references and (2) correspond to features identified in existing ontologies or databases. The lexicons, which were built from existing biomedical lexicons and human generated lists, include an extensive list of substance names with cross-references to existing lexicons and smaller lists for other features. Implemented features and sources are listed in Table 3.2. For example, the aggregatable substance lexicon was created by merging name string lists from LocusLink, RefSeq, HUGO, and SGD. Since LocusLink was recently superseded by Entrez Gene, future versions of our aggregation system will migrate to Entrez Gene. Our final lexicon includes only substance name strings associated with human and yeast genes. For more details on the construction and analysis of the feature-signaling term lexicons see (Marshall et al., 2005).

Table 3.2. Feature Lexicons Used by BioAggregate

The feature-signaling term lexicons used by the BioAggregate tagger were largely extracted from existing public sources although manually adjustments were added to correct errors or provide additional needed items (Marshall et al., 2005).

Feature Lexicon	Generation Methodology
Aggregatable Substance	Compiled from LocusLink, RefSeq, HUGO, and SGD + manual adjustments
Mutation	Manually compiled (e.g., "mutated" = Mutated, "wild-type" = Non-Mutated)
Substance Type	Manually Compiled (e.g., "protein"=Protein, "oncogene"=Gene)
Associator	Stems were identified by inspecting the most common 500 verbs extracted in relations (e.g., stem "induc" identifies induced, induces, and induce) Verbs beginning with that stem are identified as Associators
Function	*biological_function* list from the GO ontology (e.g., apoptosis) + nominalized forms of associators (e.g., induction)
Species	*common_organism* values from RefSeq (e.g., "human"=human, "baker's yeast"=yeast) + manual adjustments
Cellular Component	*Cellular_components* from the GO ontology (e.g., centriole, extracellular, and membrane) + manual adjustments
Stopword	Manually selected, common words judged to meet this standard: "ignoring this word will not mischaracterize pathway relations".

Although the tagger's functionality is entity-oriented, connectors are also processed to extract associators which characterize the relations. Using simple verb stems, associators are classified into one of four types: induction, suppression, directional association, and non-directional association. Associators can appear in a relation's connector or entity name strings.

3.5.2 Finite State Automata (FSA)

Feature lexicons are loaded into a finite state automata (FSA) to perform left-to-right, longest first pattern matching. This makes the system scalable enough to perform feature identification on more than 180,000 relations in two minutes using a 1.9Ghz processor. When a feature-signaling term is found in an item's name string, it is removed from the name string and the appropriate feature is assigned to the item. Any words

remaining after the name string has been processed are saved as a "residual". Because feature synonyms are stored in the lexicons, multiple references to the same substance or process are consolidated. Extracted features also help clarify context and granularity. Figure 3.2 highlights how decompositional tagging is applied to the entities and connectors in a relation. Longest-first pattern matching is critical, even after appropriate tokenization. For example, an entity containing the term "gene" is likely to be a gene unless the word "gene" appears as part of the phrase "gene product."

3.5.3 Aggregatable Substances

Aggregatable substance identifiers play a key role in both our aggregation and lexicon building strategies. We would generally define an aggregatable substance as a gene and its related gene products (e.g. the derived RNA transcripts and proteins). The notion implies a group of substances which are similar enough to be grouped together semantically for many analytic purposes. An aggregatable substance identifier is a unique gene identifier as recorded in LocusLink.

Previous researchers note that a gene and its related gene products (e.g., proteins) are frequently indicated in text using a shared name string (Hirschman et al., 2002), (Rzhetsky et al., 2004). This results in lexical ambiguity because a single name string (e.g. p53) can refer to both a gene and a protein. Omitting these ambiguous terms because they can refer to multiple substances would drastically reduce lexical coverage. Existing lexical resources associate name strings for genes, proteins, mRNA and other molecules with a corresponding LocusLink identifier. We leverage this data to associate a gene and its products with a single aggregatable substance identifier. Of the 100,266 distinct name

strings found in the source lists, more than 33,000 were associated with multiple substances but only one aggregatable substance.

We also use several heuristics in conjunction with our aggregatable substance lexicon. Some aggregatable substance identifiers in the lexicon are also common English words. When such a name string is encountered, a capitalization filter passes only upper or mixed case terms. Thus the terms cAMP and FOR would be assigned substance identifiers while for would not.

The aggregatable substance approach helps address the ambiguous expression problem but also has some important analytic implications. From a user/analysis perspective, a researcher studying pathways for a particular gene might also be interested in information related to the gene's products. Practically, this methodology can assign a substance identifier to a reference even when substance type cues are not available in the text. Of course it is still valuable to distinguish between genes and proteins, so we also systematically extract and record substance types during the aggregation process.

3.5.4 Pseudo-Substances

Many useful substance name strings do not appear in our lexicon because they might be newly coined, ambiguous, or left out of the ontological resources we employed. When the tagger has finished extracting all recognized feature-signaling terms, it checks for character patterns (e.g., "starts with a letter, ends with a number".) Some normalization is done (e.g., AP1 is changed to AP-1). We assign the resulting value to the entity's *pseudo-substance* feature. Examples of pseudo-substances found in our relations include stromelysin-3, Gly82, and ESR1. The name string ESR1 maps to a

pseudo-substance rather than an aggregatable substance because it is ambiguous in that it refers to both a human and a baker's yeast gene. While we would not un-ambiguously match a pseudo-substance reference to a single item in a related external resource, we can at least associate multiple occurrences of the pseudo-substance with each other. We also would like to suggest that while we do not presently use additional contextual information to resolve potentially ambiguous references, our methodology does not preclude that possibility. We have considered including this functionality either in the ARP or in the tagger, either way has implications for architecture and performance.

3.5.5 Aggregation Levels

Once features have been assigned to the elements in a collection of relations, those relations can be better organized to support analysis. Although users employing the relations in a visual interface will be able to control the details of relation aggregation, a general framework showing increasing levels of abstraction is shown in Table 3.3 (Marshall et al., 2005). These levels of aggregation will be used as defaults in the visualization interface. Table 3.4 shows the original and tagged feature versions of 2 relations extracted from PubMed articles PubMed 8985958, 8436340 and 9707425. The locus link id for the gene p53 is 7157. These relations would match differently at different levels of aggregation.

Table 3.3. Five-level Relation Aggregation Framework

While an effective analysis tool allows the user to control the details of feature-based aggregation, default levels of aggregation provide a starting point for users and for measurement of aggregation effectiveness. This framework was previously proposed in (Marshall et al., 2005). Relations become more abstract when matching rules nearer the bottom of the table are applied.

Aggregation Level	Matching Rules		Possible Applications
	Entities	Connectors	
• Baseline	complete string match		• basis of comparison
• Feature Match	all assigned features and residuals must match		• detailed pathway analysis
• Typed Substance	function or aggregatable substance and substance type must match, and residual must match	morphological forms of the same verb are combined	• pathway analysis – granularity is comparable to some human-curated databases
• Aggregatable Substance	function or aggregatable substance must match and residual must match		• explore the function of a gene and its gene products
• Simple Pathway	function or aggregatable substance must match	connector verbs are placed into one of four categories	• high level overviews and input for automated analysis

Table 3.4. An Aggregation Example

The relations below (extracted by the Arizona Relation Parser ARP) are equivalent under simple pathway aggregation but aggregation is blocked by the substance type of the p53 entities under typed substance aggregation and by the residual in the connector for aggregatable substance aggregation.

oncosuppressor gene p53 - are known to induce -apoptosis		
Agg. Substance: LL_7157 Subs. Type: Gene Residual : oncosuppressor	Associator: Induc Residual: known, are	Function: Apoptosis
p53 protein induces apoptosis		
Agg. Substance: LL_7157 Subs. Type: Protein	Associator: Induc	Function: Apoptosis

3.5.6 Comparison with other Biomedical Text Mining Tasks

Our task (decompositional feature assignment) differs from important related work such as the BioCreAtIvE competition: (1) We are focusing on organizing extracted interaction relations not tagging entity mentions. (2) We are interested in helping present pathway network results for analysis not in indexing documents to reflect the genes or proteins they contain. In last year's competition, task 1.A challenged teams to mark referent phrases in text, task 1.B was to identify a list of entities referenced in a text, and task 2 related functional annotation of proteins and function (Yeh et al., (2004)). Rather than finding reference in the text, the BioAggregate tagger organizes extracted phrases into usefully aggregatable objects. (Yeh et al., (2004)) notes that teams tried to use their systems from 1.A to support other tasks had mixed results (p.2). In addition, several BioCreAtIvE teams noted that matching the exact boundaries in the test set was difficult (p.6). Our system is intended to leverage the output of such a system in a somewhat forgiving manner. This flexibility is important because there are a number of reasonably correct ways to mark up text. In BioCreAtIvE, developers had different ideas about the correct set of markable items (p. 6) and a partial review of one version of the test set resulted in .4% change in the answer key (p. 5). This work may provide some insight into how marked phrases from various sources can be effectively processed for analysis.

3.5.7 Multiple Substance Entities

Although simple binary relations among substances are very important, relationships between compound entities are also commonly expressed in the text of

PubMed abstracts. Consider, for example, *EGF-mediated activation of Bmk1--requires--MAP-kinase kinase Mek5* from PubMed document 9790194. The first entity could be represented as a complex predicate to show the relationship between *EGF* (LocusLink id 1950), *Bmk1* (LocusLink id 5598), *mediated*, and *activation*. We will not speculate about the details as various representations are possible. In our system we actually capture up to two aggregatable substance identifiers for an entity, recording the last one found as the primary aggregatable substance. Because this kind of entity name is frequently right-headed (the main idea is on the right) we anecdotally observe that this is not a bad heuristic. We indexed this entity as substance LL_5598, associator activate, with secondary substance LL_1950. Depending upon how an application constructed the query, this relation could be found in a search related to either of the substances and matching could be performed on other relations involving the pair of entities. The analysis application would then be able to deal with the results as appropriate.

3.6 Research Testbed

We used three datasets in evaluating our system: *ARP TP53* Relations, *PROPER* entities, and *AP1* relations. *ARP TP53* is a set of 182,499 automatically-extracted relations generated by the Arizona Relation Parser from a set of 87,903 MEDLINE abstracts related to the gene TP53. The collection was created by selecting all Medline abstracts containing keywords related to TP53. Table 3.5 sows some of the relations in the *ARP TP53* dataset. Extracted relations consist of two entities, a connector and a negation indicator. The collection show very little initial overlap. We found 142,974 distinct entity names (case insensitive) and 127,397 distinct related entity pairs (ignoring

connector labels and directionality). These relations were used as we designed our system's functions and constructed our lexicons. Later, this large set was used to test the effectiveness of the tagger. Admittedly, this test set id directed at pathways related to a particular gene. However, the large size of the collection should reduce the negative impact of any test set bias.

Table 3.5. Examples of ARP output

The Arizona Relation Parser (ARP) extracts two entities, a connector, and a negation indicator for each identified relation

Original Sentences			
- oncogene mutant p53 suppresses apoptosis			
- mutant p53 blocked E1A-induced apoptosis			
- mutant p53 […] does not induce […] apoptosis			
Resulting Relations			
Entity 1	Negation	Connector	Entity 2
oncogene mutant p53	False	suppresses	Apoptosis
mutant p53	False	blocked	E1A-induced apoptosis
E1A	False	induced	Apoptosis
mutant p53	True	does induce	Apoptosis

The *PROPER* entities set includes 1.6 million entity name strings extracted by the PROPER system (Fukuda et al., 1998) from the same 87,903 TP53-related abstracts. This set is used to evaluate our system's usefulness in aggregating entities generated by systems other than ARP. Please note that PROPER does not extract relations so comparison to this data set will evaluate entity matching which is a key component of the larger relation aggregation task.

We tested the coverage of the BioAggregate tagger using 161 "gold standard" pathway relations (AP1 relations) extracted by a biologist from 50 abstracts randomly drawn from 90,773 PubMed articles related to the AP1 family of transcription factors. These abstracts were not considered during the development of the system's functions

and lexicons. We instructed the expert to select interactions between genes, gene products, and biomedical processes. We believe single expert evaluation is adequate because these experiments test feature assignment accuracy rather than relation extraction accuracy.

3.7 Experimentation

We tested the BioAggregate tagger using the three previously described datasets to address three questions:

- How frequently do various features occur in extracted relations and entities?
- How accurately do we identify those features when they occur in the relations?
- How much consolidation is accomplished in our network of automatically extracted relations?

3.7.1 Feature Occurrence Frequency

We ran the tagger on the *ARP TP53* relations, the entities extracted by PROPER, and on the manually-extracted *AP1* relations. Feature occurrence frequency is tabulated in Table VI. We found that many feature signaling terms are extracted by both PROPER and ARP. Thus, it is not only ARP entities which can be usefully decomposed. Although some features such as mutation occur infrequently (1.3-2.0% of all entities) without decomposition, such entities cannot be appropriately matched. Our approach found a substance, pseudo-substance, or function identifier in more than half of the entities we evaluated. As previously noted, we also extract associators from relation connectors.

Most (91.4%) of the connectors in the ARP relations were matched with one of the verbs in our associator lexicon.

Table 3.6. Feature Occurrence Frequencies
We found our selecetd features in many of the extracted entities evaluated.
The tagging task is relevant to both the ARP and PROPER entities.

Feature	ARP Entities	PROPER Entities
Number of Items Tagged:	364,998	1,600,223
Aggregatable Substance (e.g., *P53*)	30.1%	39.9%
Pseudo-Substance (e.g., *Gly28*)	5.9%	11.9%
Mutation (e.g., *wild-type*)	2.0%	1.3%
Substance Type (e.g., *protein*)	27.9%	17.2%
Function (e.g., *apoptosis*)	19.5%	2.3%
Species (e.g., *human*)	2.8%	1.9%
Cellular Component (e.g., *membrane*)	10.7%	2.6%
Substance, Pseudo-Substance, or Function	51.2%	52.8%

3.7.2 Feature Assignment Accuracy

To measure the accuracy of our feature assignments, we randomly selected 100 examples for each feature from the *ARP TP53* relations. Our expert checked the results by reviewing the relations and looking up items in the source documents. In a few cases, the expert was not sure if the item should have been assigned the feature. Excluding these items, feature assignment accuracy was 95% or better for all extracted features. Alternatively, if we consider the ambiguous items to be wrong, function accuracy was still 90% and aggregatable substance accuracy was 87%.

A second consideration in assignment accuracy is coverage. That is, should we have assigned features to more entities? To address this question, our expert reviewed the tagger output for the *AP1* relations. Table 3.7 compares the tagger feature assignments ("Found") to the expert assignments ("Gold Standard"). When the "Found"

assignment matches the "Gold Standard" it is considered "Correct". We report 51.4% recall for aggregatable substances. This is an encouraging result given previous results in the 30% range. However, it should be noted that various methodologies have been used in different studies to evaluate entity recognition so results are not directly comparable. We counted the number of correctly assigned aggregatable substance identifiers and divided by the total number of references to specific genes or gene products. We did not consider pseudo-substance tags to be correct; we only counted correctly associations between aggregatable substance references and LocusLink or RefSeq identifiers.

Table 3.7. Feature Assignment Accuracy
(Recall and Precision) of the BioAggregate Tagger on 161 AP-1 Relations

	Number of Items			Accuracy	
	(A) Gold Standard	(B) Found	(C) Correct	(C) / (B) Precision	(C) / (A) Recall
Entities					
Aggregatable Substance	208	131	107	81.7%	**51.4%**
Substance Type	76	75	73	97.3%	96.1%
Function	37	30	30	100.0%	81.1%
Associator	43	42	41	97.6%	95.3%
Associator Type	34	34	34	100.0%	100.0%
Mutant	6	6	6	100.0%	100.0%
Species	1	1	1	100.0%	100.0%
Cellular Component	13	13	13	100.0%	100.0%
Connectors					
Associator	177	159	150	94.3%	84.7%
Associator Type	123	123	123	100.0%	100.0%

3.7.3 Network Consolidation

While correctly labeling entity features helps capture context, index multiple references to the same substance, and connect extracted relations to external resources, we also hope to see a significant level of network consolidation as a result of relation

aggregation. Consolidated networks should result in more focused and concise knowledge representations for visualization and analysis. Because previous studies have not measured biomedical extraction systems from this perspective, we selected a variety of network consolidation measures as a baseline for future evaluation. To get an idea of how much consolidation takes place in a set of aggregated relations we chose a subset for comparison. We chose relations where:

- Every entity has an identified aggregatable substance, a pseudo-substance, or a function.
- Each entity has only one substance or function.
- The connector contains a recognized associator.

These rules control the population for potentially confounding characteristics. The filtered set includes 44,864 of the 182,499 in the *ARP TP53* dataset and might be comparable to those relations that would be relatively important for various analysis tasks.

Figure 3.3 reports the number of distinct items and relations found at each of the aggregation levels. Please note that we report Typed Substances twice: once as described in the framework and once ignoring residuals (words that were not recognized during the tagging process). We chart these separately to show the strong impact residual words had on entity aggregation. The number of distinct entities decreases somewhat at each successive level of aggregation as does the number of disjoint relations. In this analysis, a disjoint relation is one where neither entity is found in any other relation. An analysis routine would be unable to connect such a relation to other information in the network.

Table 3.8 lists a variety of network consolidation measures that might be of interest to the reader, including the distinct item measures displayed in Figure 3.3.

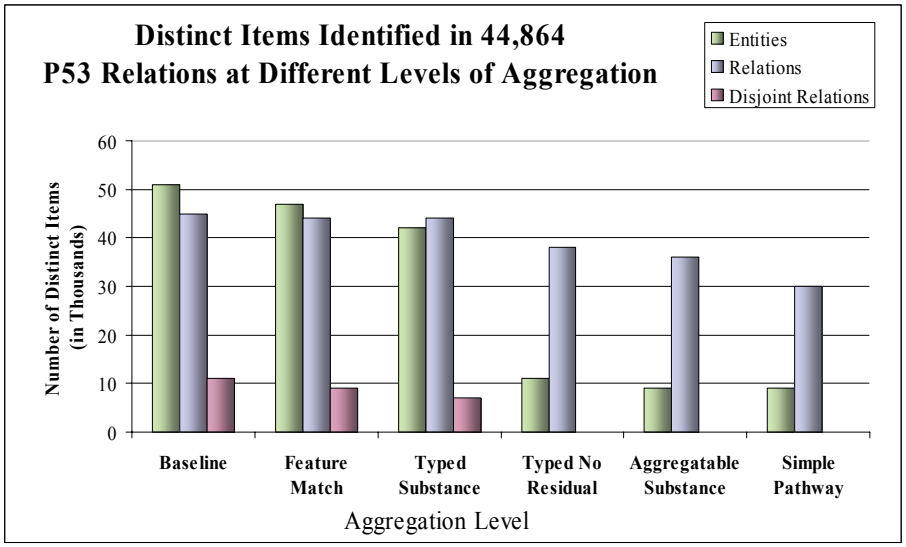


Figure 3.3. Network Consolidation

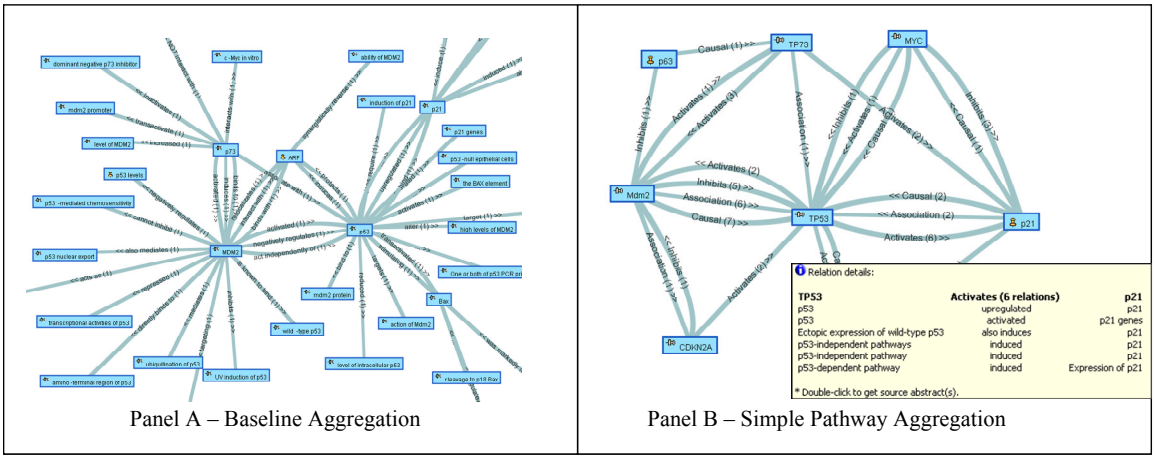
Table 3.8. Network Consolidation Measures

	Baseline	Feature Match	Typed Substance	Typed, no Residual	Aggregatable Substance	Simple Pathway
Distinct Entities	51,033	46,547	41,628	11,362	8,837	8,837
Average occurrences per entity	1.76	1.93	2.16	7.90	10.16	10.16
Distinct entities occurring 5+ times	3.0%	3.5%	4.1%	16.8%	18.9%	21.6%
Distinct relations	44,864	44,494	43,721	38,365	36,051	29,635
Average occurrences per relation	1.00	1.01	1.03	1.17	1.24	1.51
Distinct relations occurring 5+ times	0.00%	0.02%	0.08%	0.86%	1.28%	3.07%
Average number of different name strings per entity	1	1.1	1.2	4.5	5.8	5.8
Network density (linked entity pairs / possible entity pairs * 10,000)	.33	.40	.48	4.4	6.2	6.2
Number of disjoint relations	10,608	8,690	6,817	383	222	222

3.8 An Example

We queried the aggregated relations for some key substances involved in important pathways related to TP53. Panel A in Figure 3.4 shows a small part of the

unaggregated network (29 out of 277 nodes). Although many relations are displayed, it is difficult to put useful information together because the substances are represented by different strings (p53, wild-type p53, p53 levels, and transcriptional activities of p53, etc.) and redundant relations exist (p73 activated MDM2, p73 transactivate mdm2 promoter, etc.). Panel B depicts the same set of relations after aggregation. The network density is dramatically reduced. Each node in this network represents a unique gene or protein. It is easier to identify all relations between two genes or among multiple genes. For instance, it is straight forward to identify the feedback loop existing between p53 and MDM2 (TP53 Activates Mdm2 and Mdm2 Inhibits TP53). Original relations used to generate the aggregated picture are available by a mouse over, as shown in the box of Panel B for the relation of TP53 Activates p21.



Panel A – Baseline Aggregation

Panel B – Simple Pathway Aggregation

Figure 3.4. An Aggregation Example
(A) Visualization of a subset of p53 relations before aggregation (baseline level); (B) The same set of relations visualized after aggregation (simple pathway).

3.9 Disscussion and Future Directions

To evaluate extracted knowledge networks algorithms might be applied to rank the credibility of identified relations or detect apparent conflicts to chart changes of understanding over time or generate new hypotheses. Clearly, some relations are stated more than once in the literature, but they are generally stated in different terms. Aggregation allows us to consolidate. We would expect that a relationship found five or more times in the literature is likely to be "true" (that is, not the result of some extraction error and confirmed in more than one study). Table 3.6 shows that 3% of the "simple pathway" relations were found 5 or more times as compared to virtually no repetition in the "baseline" relations.

A substantial number of the relations we evaluated can be correctly mapped to Entrez Gene identifiers using our methodology. This means that a visualization system or analysis application can integrate relations from manually curated databases. Our initial observations suggest that many, or most, of the relations expressed in PubMed abstracts are expressed at higher levels of granularity as compared to those captured in manually created databases. Thus we expect that the two kinds of relations will be complementary.

Aggregation is important if we are to effectively use automatically extracted relations. The experiments and examples we report in this work suggest that decompositional aggregation is a promising methodology:

- Extracted information can be matched to external resources with reasonable accuracy.
- Networks of extracted relations can be significantly consolidated with references to the same biological object indexed at several levels of granularity.

- More concise visualizations of the same information can be created.

Our approach is intended to support flexible query responses. For example, a query related to p53 can be optionally specified to include items where p53 is know to be mutated, know to be normal, or where the mutation feature is unknown. This organizational paradigm adjusts to the ambiguity inherent in free-text sentences and NLP techniques without abandoning the possibility of detailed analysis.

Still, much additional development is needed. Thorough and detailed analysis of matching errors can be used to tune the aggregation process. For example, preliminary evaluation suggests that accounting for homologs in the lexicon construction process would further reduce ambiguity. Because genetic activity is frequently studied in a cross-species environment, this kind of indexing has promising implications. The species issue might also need to be addressed more generally. Cross-species lexical ambiguity is substantial but we use only a single aggregatable substance lexicon. Additional contextual information might be effective in identifying the species discussed in an article or referenced in a sentence. This designation could then be used to guide the tagging process to increase the number of correct aggregatable substance matches.

Other features may be both extractable and interesting; the list of tagged features can be expanded within the framework. We also plan a more extensive evaluation that addresses how accurately the relations are matched at each level rather than measuring only feature assignment accuracy and the resulting network consolidation.

3.10 Acknowledgements

This work was supported in part by Funded by: NIH/NLM, 1 R33 LM07299-01, 2002-2005, ‘‘GeneScene: a Toolkit for Gene Pathway Analysis’’ Thanks to Dr. Gondy Leroy, Chun-Ju Tseng (Lu), Riyad Kalla and the other members of the team who helped establish the early versions of the GeneScene system and interface.

4 USING IMPORTANCE FLOODING TO IDENTIFY INTERESTING NETWORKS OF CRIMINAL ACTIVITY

4.1 Introduction

Events in the last several years have brought new attention to the need for cross-jurisdictional data sharing to support investigations. As a result, a number of technology-related initiatives have been undertaken, such as the investment of $170 million in a "Virtual Case File" system for the FBI (Schmitt, 2005). Unfortunately, the system is considered dead on arrival and will likely be scrapped although significant attention is being invested in lessons learned that will probably benefit future systems. This high profile system failure highlights the difficulty of sharing investigational data across widely dispersed localities. This problem is only more difficult when multiple agencies are involved, as when local police departments have data of value to national or regional agencies. Promising computer-supported investigational models are needed to guide the development of policies, protocols, and procedures intended to increase the flow of useful information.

We are seeking to develop a methodology for identifying important investigative leads by analyzing known relationships between people, vehicles, locations, criminal incidents, and border-crossing activity. Several key challenges are clear. Administrative restrictions need to be respected and addressed; effective data integration models need to be developed; scalability issues must be carefully considered; appropriate entity matching algorithms need to be designed; and, perhaps most importantly, appropriate computer-

supported analysis models need to be proposed and tested. Analysis models are crucial because without knowing how shared data can be effectively employed, costly resources will likely be wasted in expensive but un-workable integration efforts. In addition, the volume of data that can be accessed in cross-jurisdictional data sets cannot realistically be employed by human investigators without effective filtering and selection capabilities. We are exploring cross-jurisdictional criminal activity network (CAN) analysis to support local and inter-agency investigations and operations because it promises to address some of these concerns.

Network-based techniques are commonly used in real-world investigational processes. Criminals who work together in a pattern of criminal activity can be charged with conspiracy and taken off the street for a longer period of time. While many traditional data mining techniques produce un-explainable results, criminal association networks are understandable and actionable. Many networks of associations are "drawn" only in the minds of the investigators, but visual network depictions called link-charts are commonly used in important cases. Link charts combine many cases together into an overall picture of a focused set of criminal activity. The depicted associations may be focused on particular crime types, localities, or target individuals. Other times they depict relationships in a particular case to focus an investigation, communicate within law enforcement agencies, or present data in court. Link chart creation is a manual, expensive, but valuable investigational technique.

Figure 4.1 is an example of the use of network-based information. It shows an intentionally low-resolution photograph of a link charted created in 2003 to support

investigations that involve both fraud and methamphetamine trafficking in the Tucson area. Creating the chart required six weeks of an experienced crime analyst's time. A few key targets were identified, then known associates found in the incident records of the Tucson Police Department (TPD) and the Pima County Sheriff's Department (PCSD) were evaluated to highlight other relevant individuals. The picture's low resolution is an intentional form of data scrubbing and highlights the restrictions associated with data sharing implementations and investigation-oriented research.

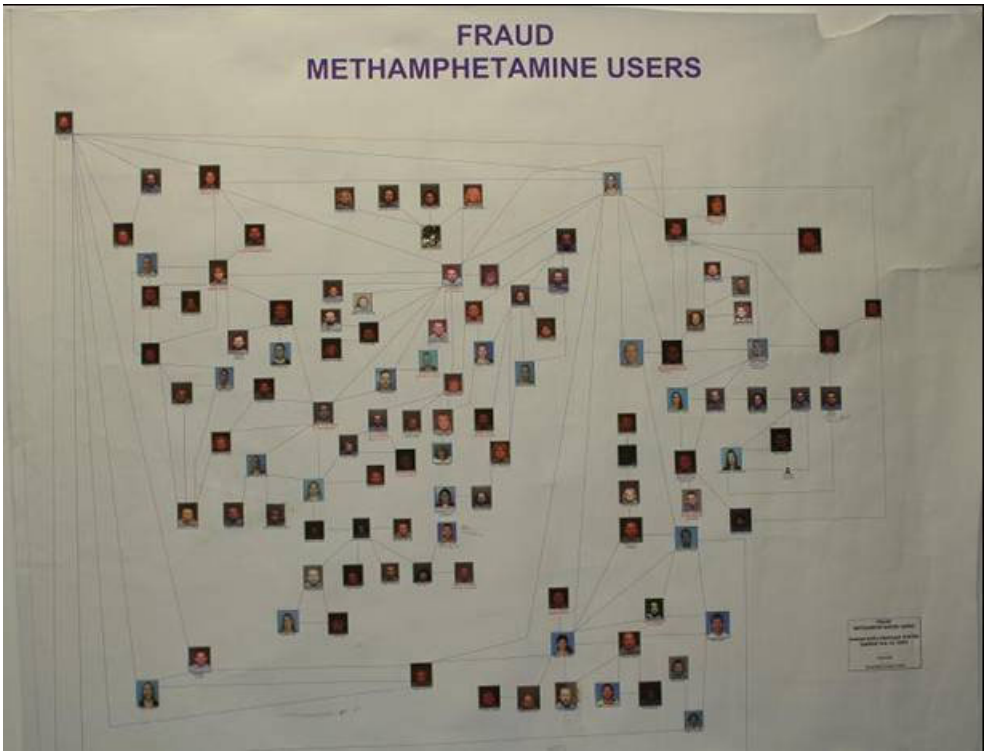


Figure 4.1. A Fraud/Meth Link Chart
This chart was manually drawn, in 2003, by an experienced crime analyst, in six weeks, using data found in Tucson Police Records (TPD) and Pima County Sheriff's Department (PCSD) records. It was intended to help with investigations involving fraud and methamphetamine trafficking.

Maturing computerized criminal records systems should provide new opportunities for network-based, computer-supported investigational analysis. For

example, most (but not all) of the information used to create the Fraud/Meth link chart was found in the computerized records of the Tucson area law enforcement agencies. If effective analysis applications were available, link chart creation would be enhanced. Figure 4.2 begins with the incident records stored by TPD and PCSD. To facilitate cross-jurisdictional link chart formation, these records have to be processed into a network of relations between people, vehicles, and locations. Computer support for this process would reduce the time (and therefore the cost) associated with link chart creation. Computer-supported link chart creation would also likely be more systematic. This would allow the processing of larger amounts of data and, although human analysis will still be crucial, it might reduce the amount of training required to create a useful chart. If useful charts could be created by more people, they might be used in more investigations.

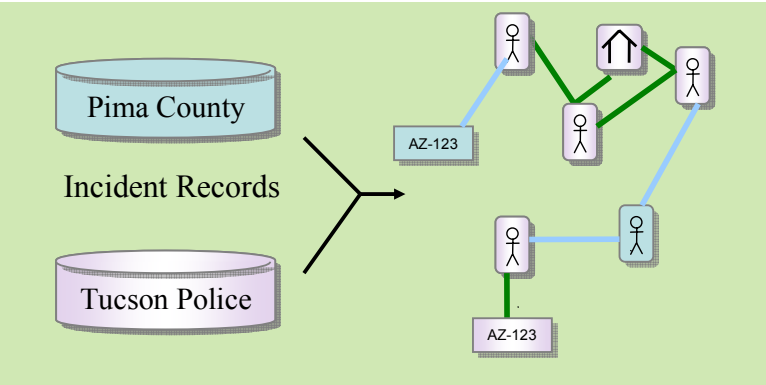


Figure 4.2. Computer Support for Link Chart Creation
In the Tucson area, computer support could be provided for link chart creation by merging existing incident records in an integrated network to identify key associations in support of investigational activities.

The idea of helping investigators create link charts supported by computerized analysis algorithms is simple to understand, but to create such a system we need to consider how criminal associations are evaluated, how incident data should be organized,

and how the appropriate items can be selected. These issues are partly addressed in the literature. Section 4.2 reviews previous research in criminal network analysis, multi-jurisdictional data integration, and interestingness algorithms. Section 4.4 describes the design concerns and implementation details of an importance flooding algorithm to assist in the selection of interesting sub-networks from large networks of illicit activity records. Sections 4.5 – 4.7 describe the testbed we used, the experiments we conducted, and the results of our investigation of the effectiveness of our algorithm. More work is needed. Section 4.8 summarizes the lessons we have learned and suggests future research directions.

4.2 Literature Review

Previous research has investigated social network analysis in the context of criminal investigations. Section 4.2.1 reviews relevant literature identifying the state-of-the-art in criminal network analysis. While previous work suggests ways to identify close criminal associations, it has not been applied in the construction of link charts. Section 4.2.2 describes our previously-published framework for integrating criminal incident data. This framework is intended to recognize the needs of criminal activity network (CAN) analysis applications and some of the difficulties associated with cross-jurisdictional data sharing. Section 4.2.3 looks at the association rule mining literature to see how previous work has attempted to identify interesting information in large data sets with an emphasis on network-based methodologies.

4.2.1 Criminal Network Analysis

Network analysis has a long history in criminal investigation (Bayse & Morris, 1987; Coady, 1985; Coffman, Greenblatt, & Marcus, 2004; Klerks, 2001). (Sparrow, 1991) highlights the importance of social network analysis techniques in this important domain identifying, a wide variety of network structure measures and logically connecting those measures with investigational implications. For example, he points out that questions such as *" 'who is central to the organization?', 'which names in this database appear to be aliases?', 'which three individuals' removal or incapacitation would sever this drug-supply network?', 'what role or roles does a specific individual appear to play within a criminal organization?' or 'which communications links within a international terrorist fraternity are likely to be most worth monitoring?' "* (p 252) would all be familiar to social network analysis practitioners. He goes on to connect specific network measures to important investigational concerns. Sparrow also considers how certain characteristics of real-life networks (size, incompleteness, fuzzy boundaries, and dynamism) are likely to impact law enforcement-related network analysis.

Some of the analysis techniques anticipated by Sparrow have been explored in more recent work. (Klerks, 2001) categorized criminal network analysis tools into three generations. First generation tools take a manual approach. That is, they allow investigators to depict criminal activity as a network of associations. A number of second generation systems have been used to visualize criminal networks including Netmap, Analyst's Notebook, Watson, and the COPLINK Visualizer. NetMap processes large collections of associations by decomposing the data into nodes and links and generating

charts that use a line's thickness or color to annotate associations between nodes (Chabrow, 2002). Analyst's Notebook from i2 supports analysis and visualization of networks of criminal activity (I2, 2004). i2 has created tools to store manually input data and map to existing external databases. Another example is KCC's COPLINK visualizer component which displays relationships and supports user drill-down to underlying details (KCC, 2004). In COPLINK, relational closeness is reflected in close proximity and levels of activity are reflected in icon size. These tools provide various levels of interaction and pattern identification.

In Klerks' taxonomy (Klerks, 2001), third generation tools would possess advanced analytical capabilities. This class of tool has yet to be deployed in organizations but several techniques and methodologies have been explored in the research literature. For example, (Coffman et al., 2004) introduces genetic algorithms to implement subgraph isomorphism and classification via social network analysis metrics for intelligence analysis. Network analysis tools to measure centrality, detect subgroups, and identify interaction patterns were used in (Xu & Chen, 2003), and the topological characteristics of cross-jurisdictional criminal networks are studied in (Kaza, Xu, Marshall, & Chen, 2005).

Shortest path measures have received particular attention. One important consideration in an investigation is the identification of the closest associates of target individuals. A variation of this analysis tries to identify the shortest path between two target individuals. These ideas, closest associates and shortest path, are clearly relevant in link chart analysis. CrimeLink Explorer employed relation strength heuristics to support

shortest-path analysis (Schroeder, Xu, & Chen, 2003). Based on conversations with domain experts, they weighted associations by:

- crime-type and person-role,
- shared addresses or phones, and
- incident co-occurrence.

An algorithm for shortest path analysis for criminal networks was implemented and tested in (Xu & Chen, 2004). Because criminal networks can be very large and very dense, the computational burden required to identify the shortest path between two individuals can be significant. (Xu & Chen, 2004) address this using a carefully crafted computational strategy.

4.2.2 Integrating Multi-Jurisdictional Law Enforcement Data

A system for law enforcement data integration must deal with data specialization, availability, sensitivity, and contextual usefulness issues (Marshall et al., 2004). In the cited work we proposed a methodology for integrating multi-jurisdictional data, identifying three classes of data and two integration steps. We aim to support cross-jurisdictional, network-based analysis considering real-world constraints and priorities. The remaining paragraphs in this section describe our data integration framework.

Combining data from independently-developed sources is a challenging task. Information integration approaches such as federation, warehousing, and mediation aim to address different needs and difficulties (Garcia-Molina & Ullman, 2002). Commonly acknowledged problems (Chen & Rotem, 1998) include (1) Name Differences: same name, different entity, (2) Mismatched Domains: problems with units of measure or

reference point, (3) Missing Data: incomplete sources or different data available from different sources, and (4) Object Identification: no global ID values and no inter-database ID tables.

The task of integrating two databases can be generally divided into two parts [5]. First schema-level heterogeneity is resolved by aligning semantically corresponding columns between the two sources. Secondly, entity matches are identified to connect objects in one database to records describing the same objects in the other database. Entity level matching is generally performed after schema-level matching is complete. The nature of the overlap between different datasets strongly affects the entity matching process. Existing matching processes can be categorized as using (1) key equivalence, (2) user specified equivalence, (3) probabilistic key equivalence, (4) probabilistic attribute equivalence, or (5) heuristic rules (Lim, Srivastava, Prabhakar, & Richardson, 1996). Variations of these approaches can be seen in existing law enforcement data sharing initiatives.

Law enforcement personnel are confident that cross-jurisdictional data sharing is important but difficult for several reasons. Most law enforcement records management systems (RMS) are not interoperable. Because of unique needs, policy implications, cultural momentum, and existing contractual arrangements it would be very expensive and organizationally difficult to get multiple jurisdictions to use compatible systems. Even if systems are quite similar in structure and function, identity records are difficult to match across jurisdictions. Finally, because vast quantities of data exist in each local system, combining several systems could reduce performance and effectiveness.

In the law enforcement domain schema-level heterogeneity is generally addressed using one of the three approaches described here.

1. Agencies allow remote access to existing systems. This first approach emphasizes connectivity. For example, the ARJIS system (ARJIS) is an extensive network accessing information from a large number of San Diego area criminal justice agencies. For these systems, data integration involves mediators that consolidate and translate queries to gather results from multiple sources.

2. Source data is mapped to other structures to support specific tools. Implementations based on this approach emphasize specific desired functionality. One example is the COPLINK system which implements an investigation-oriented database structure to support queries over law enforcement data. COPLINK was initially developed at the University of Arizona's AI Lab (Chen et al., 2002; Chen, Zeng, Atabakhsh, Wyzga, & Schroeder, 2003) and a commercial version is now developed and distributed by Knowledge Computing Corporation (KCC) (Staff, 2003). KCC creates migration routines to extract data from a client agency RMS and organize them into a flexible database structure called a COPLINK "Node." When several agencies deploy COPLINK it can be configured to support cross-jurisdictional data searching.

3. Standardized data structures are evolving to formalize the semantics of available data, as in the GJXDM project (USDJ, 2004). This approach is intended to decrease the cost and time required to implement data sharing between agencies. This is an attractive approach but an extensive set of standardized objects has not yet been widely accepted.

Even when the technical issues have been addressed, policy and privacy issues remain. Installations that host criminal activity information need to take special care to prevent unintended release of data. Investigators do not want targets of investigation to know they are being watched and smuggling organizations are known to respond quickly to changes in border monitoring activity. Thus, sharing of data between different agencies requires customized data sharing agreements (Atabakhsh, Larson, Petersen, Violette, & Chen, 2004).

Our review of existing systems and domain-specific considerations helped us develop the following framework for creating cross-jurisdictional information CANs. The key to the framework is identification of 3 classes of data: (1) base data with overlapping data from multiple jurisdictions with multiple object and relation types, (2) high volume but relatively simple supplementary data to enhance CAN information content, and (3) case specific or ad-hoc query-specific data expressing important relationships or features. Given these classes of data, integration should proceed in three steps: schema-level transformation of base data, entity-matching to align objects across data sets, and normalization and matching of supplementary data.

Base data should be semantically aligned and mapped to support CAN generation. Base data identifies associations between entities. When there is a lot of overlap between datasets, there is a lot of value to be gained. This is a classic data integration task requiring reconciliation of legacy data into a common schema and instance-level entity matching. Police RMS records are the prime example of this kind of data because multiple jurisdictions keep similar types of data about an overlapping set of objects.

Standardized data dictionaries may eventually encourage development of interoperable systems, but for now data sharing initiatives generally begin by mapping to a global schema and then move on to entity matching. Base data integration should be a repeatable transformation process so that the combined datasets can be refreshed frequently.

Entity matching in this domain will tend to rely on heuristics. Primary objects will include people, locations, and vehicles. More research into appropriate identity matching algorithms for cross-jurisdictional datasets is needed. Previous and current work in the AI Lab aims to address this issue (Wang, Chen, & Atabakhsh, 2004). Input from domain experts suggests an initial match for people using first name, last name, and date of birth. These heuristics are not perfect; a few incorrect matches may result and certainly many correct matches will be missed. Other alternatives such as FBI and state numbers may be useful but are not consistently available. Locations can be matched based on geo-codes and vehicles can be matched by license plate and/or vehicle identification number (VIN).

License plate data has some interesting and useful characteristics. Plate numbers can be recorded in an unobtrusive fashion and, while criminals frequently avoid identification by lying about their names in routine interactions with law enforcement officials, license plate numbers are directly observed. In addition, vehicles used by criminals are often registered in someone else's name. Even if a criminal uses an alias in incidents involving a particular vehicle, the resulting person-vehicle data implicitly links the incidents. License plate numbers also are occasionally transferred to different cars:

illegally when a car or plate is stolen or legally when it is sold. For many applications these characteristics make plate numbers more useful than vehicle identification numbers.

In addition to the base data, investigators use many additional supplementary or query-specific information resources to identify criminals' activities and associations. This additional data may not be readily available for a variety of reasons.

- Specialization: Frequently, useful data is not directly accounted for in the global schema. For example, police RMS systems do not usually store border-crossing events.
- Availability: Frequently, information like jail visitation histories and motor vehicle registration records are important and could be, but haven't been, included in an agency's data system.
- Sensitivity: Investigators do not want many bits of information included in widely used sources. In some cases it is feared that information would be leaked to the criminals involved. In some cases data has been subpoenaed and can be used only in a single investigation.
- Contextual usefulness: Background information and rumors identify some relationships between individual criminals, for example, "Bob and Joe are brothers" or "Fred and Jim were friends in high school." This kind of information is not collected in large quantities, applies only to specific cases, and should not be included in an RMS because of privacy and security concerns.

Our framework allows for the inclusion of this kind of data by treating it as supplementary data or as query-specific data. A data source is appropriate for supplementary integration when (1) it is available in quantity and can be appropriately

organized, (2) its sensitivity level allows for it to be shared across multiple investigations, and (3) it is contextually appropriate outside of a single investigation. Data can be used as supplementary data if it can be reduced to one or more lists of features or events directly associated with identifiable objects in the base data set. For example, mug shots of people, border crossing records, or jail visitations can all be recorded associated with particular individuals already contained in a base data set of criminal incidents. Query-specific data can be used to guide CAN building. For example if phone records indicate a suspect called 19 different people, a CAN network could query for relationships involving any of the 20 people to arrive at a more context-specific result without storing subpoenaed data in the general investigation data set. Both supplementary and query-specific data has to be normalized and matched to the objects and entities from the base data.

4.2.3 Network-Based Interestingness and Importance

Summarizing the literature review to this point, we have seen that criminal network analysis research has identified the value of network analysis in this domain, looked at characteristics of networks in the domain and difficulties that arise from those characteristics, and suggested algorithms to measure association closeness. We also described a network-based framework for combining cross-jurisdictional data. Building on this research, we want to help identify “interesting” subsets of large criminal activity networks.

The interestingness (or importance) issue is a well recognized problem in the association rule mining field. Interestingness measures seek to assign a ranking to

discovered associations based on some interestingness calculation methodology (Hilderman & Hamilton, 2001). The various measures of interestingness can be classified into two categories: objective measures and subjective measures (Silberschatz & Tuzhilin, 1996). Objective measures are generally statistical and include confidence and support. Subjective interestingness measures, on the other hand, can be classified into two groups: actionable and unexpected. (Padmanabhan & Tuzhilin, 1999) note that beliefs are important in identifying interesting associations. Results can be filtered by encoding user beliefs using some "grammar" and comparing extracted relationships to that grammar (Sahar, 2001, 2002). A way to incorporate beliefs is important for automatic interestingness analysis.

Notions of interestingness have received special attention in the context of data that can be represented as a network. Some researchers emphasize that interestingness is relative. For example, a "root set of nodes" within a larger network are used to enhance relevance searching in (White & Smyth, 2003). They describe a general class of algorithms that use explicit definitions of relative importance. The two main intuitions behind the approach are that 1) two nodes are related according to the paths that connect them, and 2) the longer a path is, the less importance is conferred along that path. They employ a scalar coefficient to "pass" smaller amounts of importance to nodes as the distance between a pair of nodes increases. They note several ways of choosing non-overlapping paths between node pairs. These notions of relative importance align well with the cognitive model described by investigators we have talked with. To identify

associates or leads, investigations begin with some target suspect and look for the closest associates.

(Lin & Chalupsky, 2003) detect novel network paths (not just nodes or links) to reveal interesting information. This was a novel way of analyzing the HEP-Th bibliography dataset from the Open Task of the 2003 KDD Cup (Gehrke, Ginsparg, & Ginsparg, 2003). They evaluated bibliographic citation data to answer questions such as "which people are interestingly connected to C.N. Pope?". The basic notion of their analysis was to detect interesting short paths through a network rather than to detect interesting nodes. They categorized link types and used multiple node types in their network. So, for instance, universities were associated with authors who had published a paper while affiliated with the university, and authors were associated with their co-authors. Without putting in specific rules defining "interesting" their algorithm discovered that Mr. H. Lu. was the most interesting person relative to C.N. Pope because he interacted with Pope along a variety of network paths. These paths take the following form:

- [Lu]-writes-[Paper1]-cites-[Paper2]-written_by-[Pope]
- [Lu]-authors-[Paper1]-authored_by-[Pope], and
- [Lu]-authors-[Paper1]-authored_by-[Person1]-authors-[Paper2]-authored_by-[Pope].

This notion that interestingness is path-based rather than node-based is applicable to criminal investigations. For example, the analyst who created the Fraud/Meth link chart noted that she was more interested in people who sold drugs and were associated

both with people who sold methamphetamines and people who committed fraud. This kind of association pattern is a short path through the criminal activity network.

4.2.4 Design Goals

Previous methodologies such as criminal network analysis and shortest path evaluation have not been used to address the important task of link chart creation. Previous criminal network analysis research has focused on a social network approach which considers only association strength, ignoring other indicators of contextual importance. The association rule mining literature suggests several approaches intended to identify interesting items in networks. Our goal is to combine and adapt criminal network and interestingness techniques to support investigational tasks while allowing for the real-world challenges of this important domain. If effective, we expect such a methodology to be useful in a variety of real-world network evaluation applications.

We plan to use police incident records as the base data for our analysis methodology. Conversations with crime analysts and investigators have highlighted some important considerations. When only law enforcement incidents are used, personal associations (e.g. family) may be missed. Key individuals can appear un-important until caught committing a serious crime and obscure individuals can be important if they link investigational targets. Links are difficult to assess. Very different associations may look the same in the records or be more or less important in context. Inquiry should be target focused. Resource allocation and privacy concerns discourage “fishing” for new targets.

Based on our review of the literature and our conversations with investigators we developed a list of design goals:

1. Allow query-specific information to fill in missing data. It is clear that some associations (e.g. boyfriend/girlfriend relationships) are considered important but are not generally captured in police incident records. An effective methodology should be able to incorporate this kind of information in the analysis process in a more substantive way than simply allowing it to be depicted in the final link chart.
2. Incorporate domain-appropriate heuristics (or beliefs) to support analysis. Different investigations will have different focuses. For example, the fraud division may not be targeting active, dangerous criminals who are not involved in fraud.
3. Encode these heuristics in a format that can be adjusted at query time for new insights. As an investigation progresses, new insights arise. For instance, if the fraud unit realizes that many fraud cases are related to methamphetamine trafficking, they might seek to re-analyze data with an emphasis on this important correlation.
4. Tolerate missing and ambiguous data. As cited in previous literature (Sparrow, 1991), and obviously, criminal records are incomplete. While missing information will always be expected to reduce the effectiveness of an analysis mechanism, a good methodology for this domain needs to be somewhat tolerant of ambiguous network representations.
5. Be target focused. Several large-scale, government-sponsored data mining initiatives have been severely criticized because they did not adequately allow for individual privacy. We want to use a methodology that conducts analysis based on associations with established individuals not a general selection routine that

"fishes" for bad guys in public records. This goal is also important because investigational resources are limited and investigational assignments are distributed. A particular investigator or team of investigators needs to be able to focus attention on specific targets.

Importantly, these goals are applicable to smaller local investigations but are also relevant to large-scale inter-jurisdictional investigations. And, although the justification we provide for them is explained in a law enforcement context, these same goals would be useful in guiding design of analysis techniques applicable to other network evaluation applications.

4.3 Research Question

Within the broader context of supporting investigations by generating useful leads, this essay studies a methodology for increasing the efficiency of link chart creation to (1) save time and money, (2) allow the technique to be used in more investigations, and (3) employ large quantities of available data. Such a model can be used to support investigations and to guide the implementation of data sharing systems. The research focus of this essay can be summarized in a single research question:

How can we effectively identify interesting sub networks

- useful for link chart creation
- from associations found in a large collection of criminal incidents
- employing domain knowledge

- to generate useful investigational leads and support criminal conspiracy investigations?

4.4 System Design

4.4.1 Architecture

The CAN visualizer under development at the AI Lab at the University of Arizona is intended to be a 3rd generation network analysis tool. It supports automatic clustering of criminals into groups, it calculates centrality and betweenness values to help identify leaders or gatekeepers, and it is built to use data integrated from multiple jurisdictions. The algorithm we develop and test in this work can be included in the tool so that analysts and detectives can interactively construct investigational link charts. The work presented here represents one phase of the larger process depicted in Figure 4.3. Police records from two local jurisdictions have been converted into a common schema. Records for people with the same first name, last name, and date of birth are matched to form a network of incident-based associations. Starting with a target list of suspects and sets of link weight rules and importance heuristics, individuals are importance ranked for inclusion in investigation-specific link charts.

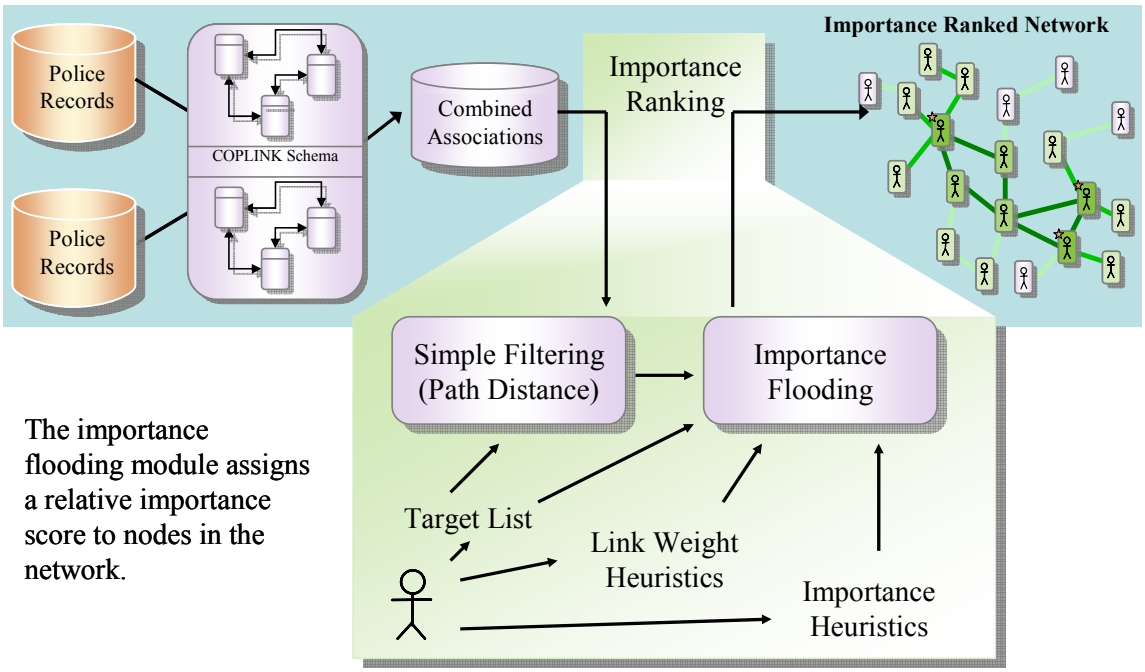


Figure 4.3. Identifying Interesting Sub-networks of Criminal Associations
Incidents are extracted from police records and organized into networks of criminal activities. Leveraging the resulting path structure, a list of target individuals, and user defined heuristics, the importance flooding algorithm produces an importance-ranked network to support creation of investigational link charts.

4.4.2 Importance Flooding Algorithm Overview

We propose the use of an importance flooding algorithm to identify interesting sub networks within larger networks. It was designed to allow input of associations not known in the base dataset, search based on initial target individuals, and implementation of inquiry-specific importance heuristics. The basic intuitions of the algorithm are (1) associates of interesting people become relatively more interesting and (2) both a person's past activity and their involvement in interesting association patterns establish initial importance. The algorithm considers two key network elements in its calculation (1) association closeness and (2) importance evaluation. The calculation leverages association closeness measures as suggested by (Schroeder et al., 2003), scalar

coefficients as in (White & Smyth, 2003), and leverages a path-based notion of interestingness reminiscent of the methodology used in (Lin & Chalupsky, 2003).

This approach differs from previous work in this domain in several ways. (1) It is applied directly to the task of link chart generation. Although previous work has hinted at this kind of application, it has not been directly applied or tested in that context. (2) We combine both structure (closeness-weighted associations) and activity-based contextual importance heuristics (e.g. "we are looking for people who have been involved in fraud") in our computation where previous work focuses on social network measures that consider only the structure provided by closeness-weighted associations. (3) We encode the users' importance notions or "beliefs" as short paths through the network. In some cases this is simple grouping (e.g. people who have been suspects in fraud incidents) but we also systematically process patterns of relationships. For example, one of the heuristics we use in our testing process captures the analyst's input that she was more interested in people who sold drugs and were associated both with people who sold methamphetamines and people who committed fraud. This algorithm favors individuals who are involved in user-defined interesting paths.

4.4.3 Importance Flooding Algorithm Details

The algorithm proceeds in four basic steps.

1. Weights are assigned to network links.
2. Initial importance values are assigned to network nodes.

3. Importance is passed (flooded) from important nodes to nearby associates. This results in a final importance score for each node.
4. A network subset can then be selected starting with investigational targets using a best first search paradigm that considers the scores from Step 3.

For purposes of illustration, we will consider some large network of criminal incidents. This network consists a (A) a set of nodes, (B) a set of associations such that each association connects two of the nodes and is described by a set of properties, and (C) a set of relation weights consisting of a single link weight for each unique pair of nodes connected in the associations. In addition to the network we have two sets of rules, link weight rules and initial importance rules. Finally, the algorithm is controlled by a set of starting nodes (a subset of all the nodes in the network) and a decaying distribution function or scalar coefficient.

The nodes in the network we test in this work are all individuals although the algorithm could also evaluate location or vehicle nodes. The properties we consider in this work include crime type, from role (the role of the first of the two nodes in the association), to role (the role of the second node in the association), and crime date. This kind of network can easily be generated from many criminal records systems. It does not require analysis of items such as MO (modus operandi) or physical description. What's more, our current representation categorizes crimes using standard crime types which do not differentiate, for instance, between drug crimes involving methamphetamines vs. drug crimes involving heroine or marijuana. Certainly these features can play an important investigational role but extraction of such details would likely be expensive,

inconsistent, and subject to additional administrative and privacy restrictions in a cross-jurisdictional environment. Our goal was to test our methodology using data that would be relatively easily to share. With all that being said, additional features could be used by the algorithm simply by changing the initial input rules. The main computations in Steps 3 and 4 would be unaffected by inclusion of the additional information.

4.4.4 Assigning Link Weights

Previous research has made it clear that association strength is an important consideration in criminal network analysis. Further, it has been suggested that the association strength is a function of the roles two individuals played in an incident and the frequency with which those individuals "co-occur" in incident records. We accept the value of these heuristics, but other rules may also be appropriate. Thus, while our methodology generally allows for any link weight assignment technique to be employed, our current version accepts an XML file which allows a user to rate the strength of an association using its various properties. Relation weights range from 0 to 1. In general, relation weights are assigned to node pairs by evaluating the corresponding associations as a function of the number of associations and properties of those associations.

We used relatively simple heuristics in the testing presented in this essay. For example, our rules expressed a strong relational weight for a pair of individuals who were both recorded as arrestees in the same incident, but a lower weight for associations where the two individuals were considered investigational leads in the same incident. In addition to these initial link weights, frequency of association was considered. (Schroeder et al., 2003) found that when a pair of individuals appears together in four or more police

incidents a very strong relationship exists. We therefore assign a relation weight of 1 to any node pair occurring together in four or more incidents regardless of crime role or incident type. When less than four incidents connect two individuals, we multiply the strongest association weight by 3/5, the second strongest by 1/5 and the third strongest by 1/5 and sum the products. The 3/5, 1/5, 1/5 distribution is somewhat arbitrary but it is reasonable in light of previous research and implements a methodology for balancing strength of association with frequency of association. Intuitively we would say two individuals are more closely related when they have been co-suspects in three incidents as compared to one or two.

4.4.4.1 Initial Importance

In addition to the association closeness weights, initial importance values are assigned to nodes using path-based importance heuristics. In general, importance weights could be assigned a number of ways. In our current implementation, we accept three kinds of importance rules which are loaded from an XML file: (1) Simple activity-based group rules, (2) multi-group membership rules, and (3) path rules. Figure 4.4 describes the three types of rules. Weight values are assigned to each rule, each node is evaluated for group membership based on the rule, and a node is assigned an initial importance score equal to the sum of the weights of all groups to which the node belongs. The link weight and importance values assigned in our implementation were derived from previous research or developed in conversation with crime analysts and require only information that is likely to be available in a cross-jurisdictional setting. However, many

other variations of link weight or initial importance could be used by the importance flooding algorithm.

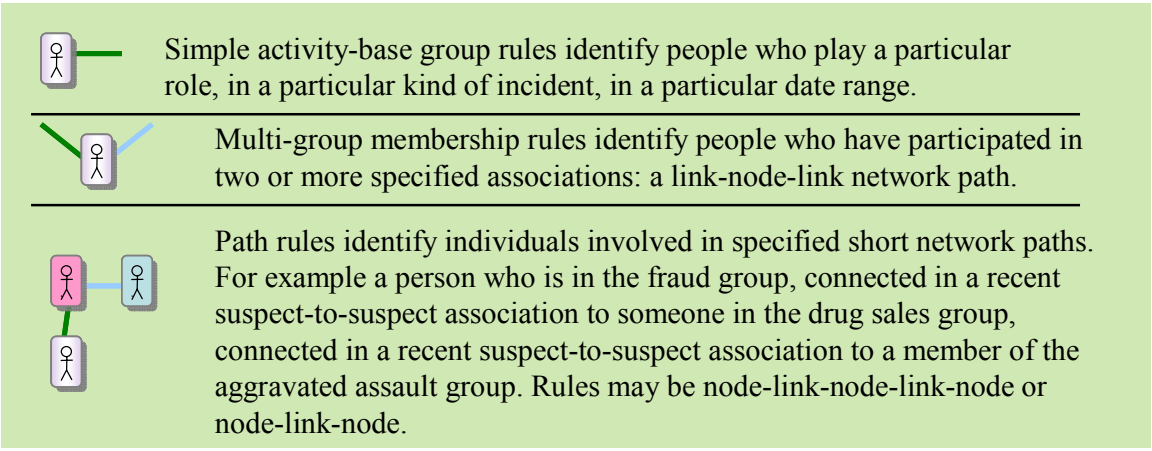


Figure 4.4. Three Types of Initial Importance Rules

First we have simple group membership. For each node, we evaluate the associations for that node to see if it contains any that meet the rule criteria. Rules can generally test for any properties in the link. In our testing we used rules that test the role of the node in the association, the crime type of the association, and the crime date. Each of these weight rules also contains an importance value. We used values from 1 to 5 but most any positive values could be used. In general rankings would be set based on the analyst's judgment. For example, if recent fraud incidents where a person was a suspect were thought to be approximately twice as important as five year old drug arrests then values of four and two respectively could be used.

Multi-group and path rules specify short network paths which confer interestingness upon included elements. A multi-group rule simple requires that a node be a member of two or more other groups. This could be visualized as a link-node-link network path. Longer node-link-node and node-link-node-link-node paths combine group

membership and link rules. Rules can be specified using any of the properties recorded for an association. Figure 4.4 includes path-rule examples. If a node participates in a path described by a multi-group or path rule, the importance value for that rule is added to the initial importance score of the node.

4.4.4.2 Importance Flooding

The initial importance values and link weights are used in conjunction with the starting node list and decaying distribution function to pass (or flood) importance from ranked nodes to nearby associates. The algorithm itself is really quite simple. It uses:

- a set of nodes N,
- a set of weighted relations R connecting unique pairs of nodes from N,
- a set of target nodes T where all members of T are in N, and
- a decaying distribution function D which specifies the rate at which importance is passed over a network path of a given length.

Pseudo code for the importance flooding algorithm is shown in Figure 4.5. The calculation can proceed in multiple iterations creating a kind of flooding action to pass importance from identified nodes to nearby associates along associations in the network. The number of iterations could be determined several ways. In the tests presented in this work we somewhat arbitrarily set the value at 4. This would allow some importance to travel to nodes 8 hops away given a decaying distribution function with a maximum path length of 2. Instead of using a fixed number of iterations, the algorithm could be set to terminate using some convergence criterion such as minimum change threshold. We do not consider our current dataset to be adequate for testing possible convergence criteria.

In each iteration, new importance values of the nodes in N are derived and normalized to a value between 0 and 1. Target nodes are reinforced so that they are always ranked as the most important nodes in the network. Reminiscent of the similarity flooding algorithm presented in (Marshall et al., 2005) the initial importance score for a node is also reinforced in each iteration. In both schema and concept map matching applications this algorithm variation increased accuracy and reduced time to convergence. Attention is given to avoiding loops by ensuring that a node is not already in the traced path.

The decaying distribution function D is a scalar coefficient specifying a multiplier to be used as importance is passed along paths in the network. For example, we used [.5, .25] as our scalar coefficient in this work. This means that the importance value for a node is multiplied by .5 before it is passed to directly adjacent nodes and by .25 before it is added to nodes 2 hops away. This value is comparable to the value used in (White & Smyth, 2003) and in previous work they cite. While longer paths could be used (and we did experiment with some) the iterative property of the computation allows importance to be passed over longer "distances" without incurring the significant computational overhead that could result from a long decaying distribution function. To illustrate this point, consider a network with 100 nodes and three associates for each node. If a scalar coefficient of [.5, .25] is used in two iterations, importance is passed to nodes three hops away and ((100 * 3) + (100 * 32)) * 2 = 2,400 passing computations are performed. On the other hand, if a scalar coefficient of [.5, .25, .125] is used and only one iteration is performed, importance is still passed to nodes that are 3 hops away but 2,900 = ((100* 3)

+ (100 * 32) + (100 * 33)) computations are performed. Longer coefficient paths increase computational cost exponentially.

```
Each node has: unique identifier "ID," initial importance score "INIT," previous importance score "PREV,"
                and an accumulated amount of importance added in this iteration "ADD"
The algorithm proceeds using a main loop and a recursive path tracing method.

A maximum node importance score of Init+Prev+Add "MAXVAL" is maintained for each iteration as each
node and path is processed. This score is used to normalize the values at the end of each iteration.

Decaying Distribution Depth "DDD" is 2 when two values are included in the scalar coefficient
(e.g. [.5, .25]).

Main Process:
// Initialize
For each node N1
        N1.PREV= 0,  N1.ADD = 0
For each iteration
    // Call the recursive path tracing method
    For each node N1 in N
        PassAmount = N1.PREV + N1.INIT
        PathList = N1.ID
        PathLength = 1
        recursivePathTrace (PassAmount, PathList, PathLength)
    // Normalize and re-initialize
    For each node N1 in N
        N1.PREV = (N1.PREV + N1.INIT + N1.ADD) / MAXVAL
        N1.ADD = 0
    // reinforce investigational targets
    For each node T1 that is a member of the TargetNode List
        T1.PREV = 1

Recursive Path Tracing Method:
recursivePathTrace (PassAmount, PathList, PathLength)
    PassingNode = The last node included in PathList
    NumOfAssociates = The number of nodes associated with PassingNode
    For each node Na associated with the PassingNode
        if Na is not already included in PathList
            RELWGT = the relation weight for the pair [PassingNode,Na]
            DECAYRATE = the decay coefficient for paths of length PathLength
            PASSONAMOUNT = PassAmount * RELWGT * DECAYRATE * (1 / NumOfAssociates)
            Na.ADD = Na.ADD + PASSONAMOUNT

            if PathLength < DDD
              recursivePathTrace (PASSONAMOUNT, PathList + Na.ID, PathLength + 1)
```

Figure 4.5. The Importance Flooding Algorithm

4.4.4.3 Best First Search Selection

The final importance scores assigned for the network nodes are used to expand the network from the target nodes to a network of some specified size. The nodes in the starting list of target nodes are placed into a list of visited nodes and into a priority queue with a priority value of 2. Nodes are sequentially popped from the queue until all nodes have been selected or until the application has reached some maximum number of selected nodes. As each node is popped, the algorithm adds it to a list of selected nodes and then searches for all other nodes associated with that node. If the associated node is not already in the visited node list, it is added to the priority queue with its importance score (which can range from 0 to 1) as its priority value. Intuitively, the algorithm asks: of all the nodes attached to any of the selected nodes, which has the highest importance score? An analyst using the output might well consider which node to add to a link chart next using this procedure.

4.5 Research Testbed

To test the effectiveness of the proposed algorithm we needed:

- a large collection of criminal association records,
- example link charts,
- a set of heuristics rules for link weight estimation, and
- a set of heuristic importance rules.

The testbed of associations used in our experiments was prepared using the framework described in Section 4.2.2. Base relations were extracted from incident

records from the Tucson Police Department (TPD) and the Pima County Sheriff's Department (PCSD). Whenever two people are listed in the same police incident, an association record is created with "crime type," "from role" (the role of the first of the two nodes in the association), "to role" (the role of the second node in the association), and "crime date" properties. Practitioner-suggested guidelines were used to match the people found in the records. Individuals across the data sets were matched when they had the same first name, last name, and date of birth. We are quite sure that some correct matches were missed due to data entry errors or intentional deception but, in the absence of a better methodology, this matching approach seems reasonable for large-scale analysis. The combined set includes records from 5.2 million incidents involving 2.2 million people.

We obtained access to two large link charts prepared for the TPD fraud unit. The first depicted key people involved in both methamphetamine trafficking and fraud as described in Section 4.1. The chart includes 110 people and took 6 weeks to create. The second was related to an investigation referred to as "Arrow Key." The final printed link chart from the Arrow Key investigation is shown in Figure 4.6. In respect of privacy and security restrictions, only limited case details are included here.

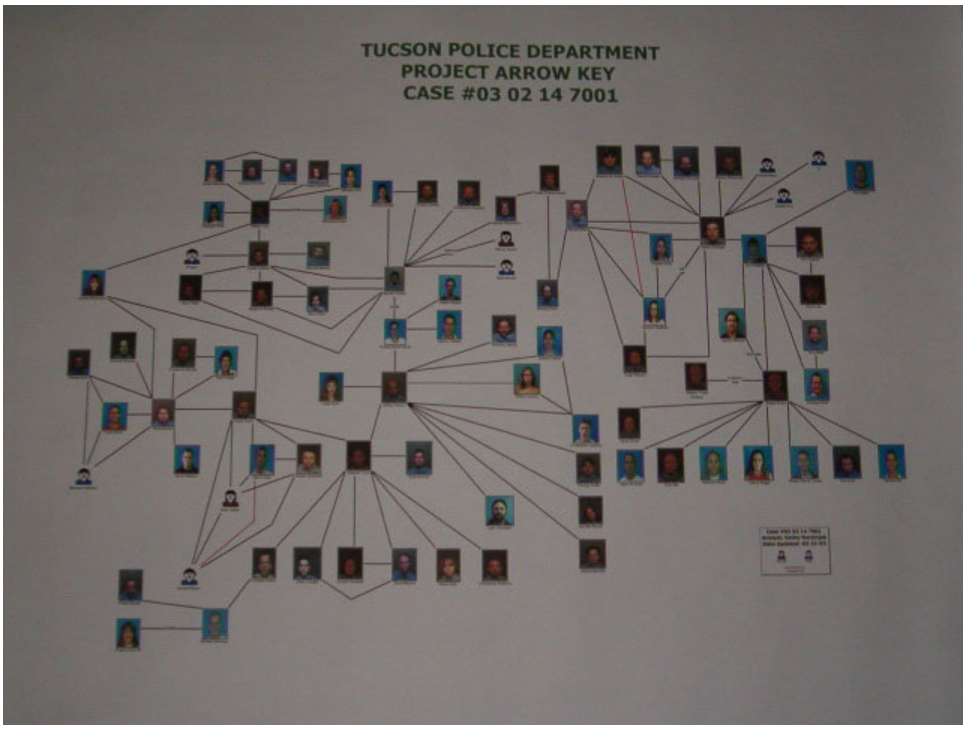


Figure 4.6. The Arrow Key Link Chart
The arrow key link chart was drawn starting with 23 target individuals. It depicts associations between 110 people.

The heuristics we used came from two sources: previous research guided the development of the very-general link weight heuristics and case priorities dictated the importance rules. The rules we used are listed in Figure 4.7.

- Link Weight Heuristics
  - Suspect/Suspect Relationships = .99
  - Suspect/Not Suspect = .5
  - Not Suspect/Not Suspect = .3
- Frequency Adjustment
  - 4 or more associations, weight = 1
  - else, ∑ (strongest relation * .6, 2nd * .2 , and 3rd * .2)
- Importance:
  - Groups: Aggravated Assault (A), Drug Sales (S), Drug Possession (P), Fraud (F)
    - (A),(D), or (F) = 3
- Path Rules: (all applied only to crimes after 01/01/2001)
  - Nodes with any 2 (A),(D),(F) = 3, (A),(D) & (F) = 5
    - (A)-(D)-(F) = 5
    - (A)-(D), (A)-(F), (D)-(F), (P)-(F) = 3

Figure 4.7. Association Closeness and Importance Heuristics

In studying each link chart we first chose a large "universe" of individuals that had potential relevance to the case. To approximate the search space considered by the human analyst, we include only people within 2 associational hops of the targets. Investigators tell us they are generally not interested past that limit. For obvious reasons, we also limited our "universe" to records that were present in the police records prior to the date the manual link chart was created. This filtering process resulted in 4,877 individuals for the fraud/meth investigation, including 73 of the 110 "correct" individuals depicted in the manually created link chart. The Arrow Key universe included 6,025 people and 100 of the 110 charted individuals.

4.6 Experimental Design

We compared our algorithm's results to the human drawn link charts. To see if it was effective, we considered how it might impact the effectiveness of time spent working

on the link chart. When an analyst creates a chart, they begin with one or more target individuals, look for associations involving those individuals, and evaluate each potential associate to see if they are important enough to be included in the chart. Starting with only a few individuals, an analyst can quickly arrive at thousands of potential associates. If the analyst could review more promising potential associates first, they would likely create a good chart in less time. The basic goal of our testing was to start with the same information considered by the human analyst and produce an ordered list of individuals such that selecting them in order forms a network. Selection methodologies that listed the "correct" individuals earlier in the list were considered to be "better." We compared several methods of ordering the list. All the methodologies resulted in a target-focused network of associations. That is, a list was generated starting with identified investigational targets and every individual added to the list had to be directly associated with someone already in the list.

- Breadth First Search: This was used as a baseline for comparison. Start with the target individuals and choose everyone directly connected first, then choose everyone two hops away, and so on.
- Closest Associate: This is a link-chart application of previous studies on shortest paths in criminal activity networks. New individuals are added to the network in order of association closeness to someone already included in the network.
- Importance Flooding: The importance flooding algorithm was used to rank all the individuals in the universe. New individuals are added to the network using this

selection rule: Choose the highest ranked individual associated with any of the people already included in the network.

- Perfect Importance Flooding: This was measured to establish a "best-case" scenario. The notion was to see how well we could select the charted people using the importance flooding paradigm if the heuristics used were completely accurate. If some general heuristics were able to completely distinguish correct nodes from incorrect nodes, would our methodology be able to effectively employ those heuristics? Because investigators certainly consider the past activities of potential suspects, it seems unlikely that an association-closeness methodology, as compared to a methodology which uses both association closeness and importance heuristics, would ever be able to accomplish "perfect" selection. To simulate perfect heuristics, we fed the importance flooding algorithm the correct answer by reinforcing the correct nodes to a normalized value of 1 after each iteration.

In addition, hypothesizing that the importance flooding methodology is useful, we wanted to explore which factors contributed to the algorithm's success. So we considered two additional approaches:

- Path Heuristics with No Flooding: In this case we used the path-based heuristics to rank importance but we did not flood that importance to nearby nodes. This was intended to show that both the initial importance of a node and its structural place in the network impact its chart-worthiness.

- Node-only Importance Flooding: This case is intended to demonstrate that the path-based heuristics add to the algorithm's effectiveness as a supplement to node-only analysis.

To form a set of hypotheses for comparing the ordering methods, we needed a numeric measure of effectiveness. We expect that on average, in networks ranked by a "better" algorithm, an analyst would have to look at fewer listed nodes before finding a "correct" node. So for analysis, we consider a measure *A* which operates for a ranking method (technique) over a size range. As each node is added to a network, we can compute the total number of nodes added divided by the number of "correct" nodes added. This ratio computes the number of nodes an analyst would have to consider for each correct node considered. A smaller number is better in that the analyst would have spent less time on un-interesting nodes. Our measure A is the average of the ratio over a range. For example, consider *A (importance flooding) at 250* = average ratio of selected nodes to "correct" nodes, selected by the importance flooding algorithm, when the number of selected nodes is 1,2,3…250. We developed the hypotheses shown in Table 4.1.

Table 4.1. Hypotheses

Techniques: • BFS = breadth first (rank by # of hops) • CA = closest associates • IMP = importance flooding • PATH = path heuristics, no flooding • NO = only node heuristics, flooding • PIF = perfect information flooding
All techniques improve on BFS • H1a: A(IMP) < A(BFS) • H1b: A(CA) < A(BFS)
Importance flooding outperforms closest associates • H2: A(IMP) < A(CA)
Importance flooding outperforms heuristics with no flooding • H3: A(IMP) < A(PATH)
Path heuristics improve on node only heuristics • H4: A(IMP) < A(NO)
Given "Perfect" information outperforms other techniques • H5a: A(PIF) < A(IMP) • H5b: A(PIF) < A(CA)
These hypotheses are expected to hold when 100, 250, 500, 1000, and 2000 nodes have been selected.

4.7 Results

The performance results for the four basic selection methods (breadth first search, closest associate, importance flooding, and perfect importance flooding) are reported for the Fraud/Meth and Arrow Key investigations in Figures 4.8 and 4.9 respectively. In the Fraud/Meth results the importance flooding approach consistently found more of the correct nodes for any given number of nodes selected. The breadth first search method did the least well. The closest associate method seems to have generally outperformed breadth first search. And, as expected, perfect importance flooding resulted in very accurate selections. It selected the correct node every time until there were no more directly-linked correct nodes. Then a few "incorrect" nodes were selected creating a bridge to the remaining correct nodes. The results were similar for the Arrow Key investigation except that the breadth first method fared somewhat better than it did in the Fraud/Meth data. This difference is not all that surprising because a much larger set of starting nodes was used.

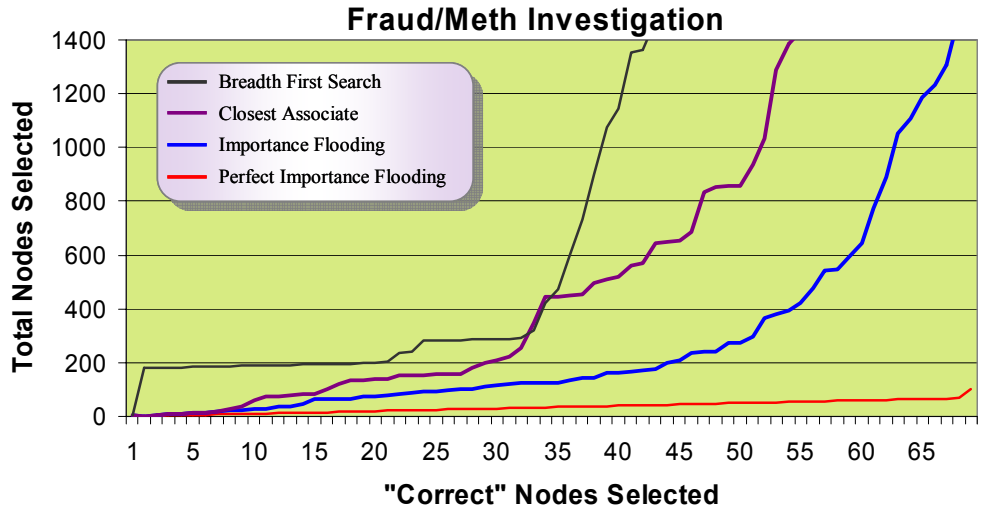


Figure 4.8. Comparison of Ranking Methods for the Fraud/Meth Link Chart
The importance flooding algorithm (blue) consistently outperformed both the breadth first search and closest associate methods. The perfect importance flooding method selected exactly the right nodes except in a few cases where no direct link existed.

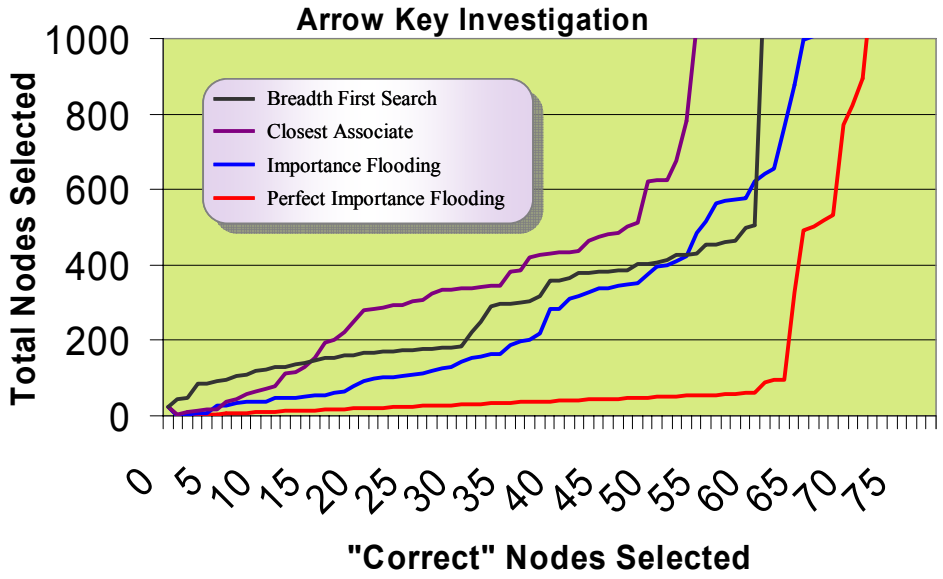


Figure 4.9. Comparison of Ranking Methods for the Arrow Key Link Chart

To expand the comparison of the importance flooding and closest associate approaches and to test the sensitivity of the algorithm to different target input nodes, we

selected three alternate sets of four starting nodes each. We did this by visually inspecting the link chart and picking nodes that were reasonably well connected to other nodes but not included in the target set of either the original investigation or any of the other alternate sets. Figure 4.10 compares the importance flooding and closest associate results for the original starting nodes and the 3 alternate sets of nodes. Again, importance flooding consistently outperformed closest associate.

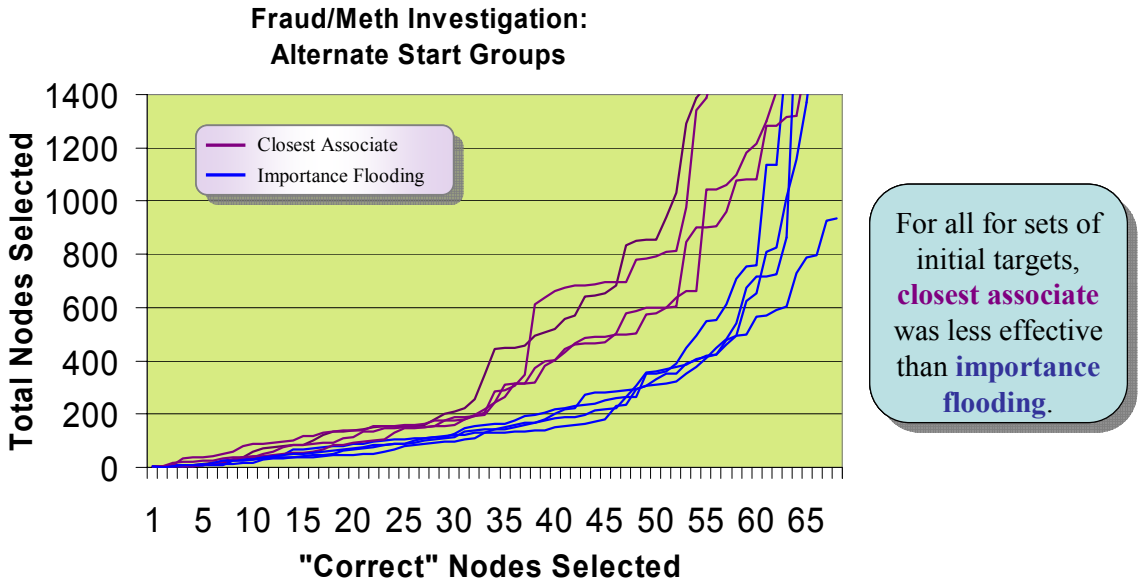


Figure 4.10. Alternate Starting Node Evaluation

We also performed numeric tests to evaluate the statistical significance of our results and evaluate the success factors driving the importance flooding algorithm's performance. Figure 4.11 graphs the results. This analysis and the hypothesis test results shown in Table 4.2 were based on the Fraud/Meth link chart data.

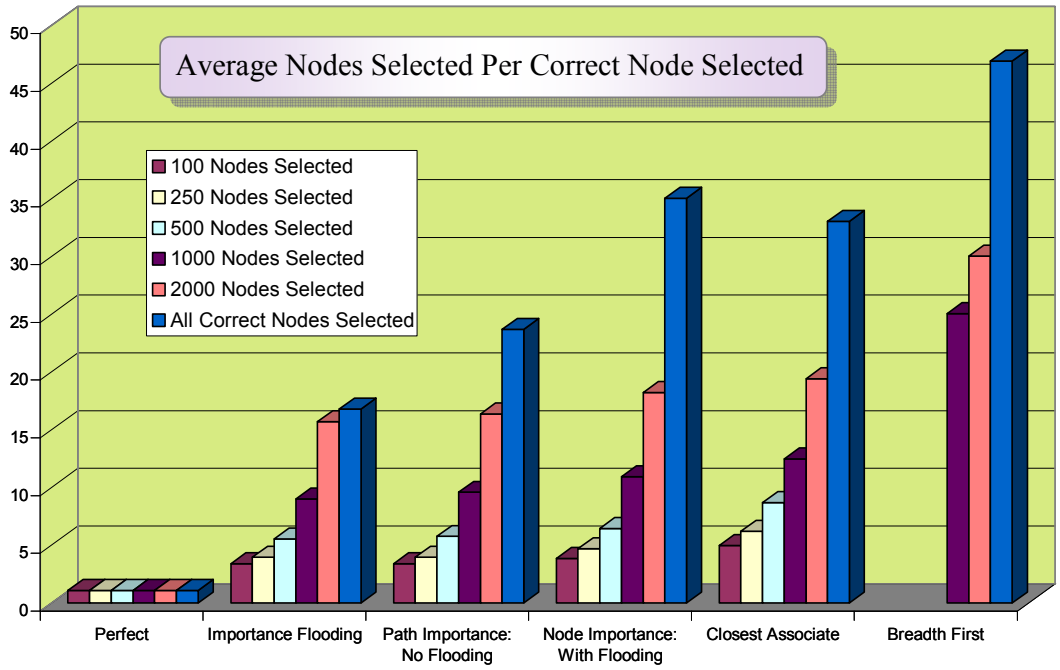


Figure 4.11. Node Ranking Methods Compared
The average nodes selected per correct node selected is an indicator of how many "incorrect" nodes an analyst would have to consider to get one "correct" node. As more nodes are selected, a higher proportion of "incorrect" nodes have been included.

Table 4.2 reports the hypothesis testing results, while Table 4.3 reports the mean, and standard deviation for each test. The sample size for each test was based on the number of selected nodes: 100 samples at the 100 node level, 250 at the 250 node level and so on. We tested 4 hypotheses. Hypotheses H1a and H1b were accepted, suggesting that both the closest associate and importance flooding methods performed better than the baseline breadth first search approach. The acceptance of hypothesis H2 demonstrates that the importance flooding method outperformed the closest associate method. The acceptance of H3 at the 500, 1000, and 2000 node levels suggests that the passing of

importance to nearby associates improves on the initial importance scores derived using the path-based importance heuristics alone in larger sets of selected nodes. Hypothesis H4 was accepted implying that our use of path-based heuristics improved on the use of only node-based heuristics. This combined with the very similar performance levels observed for importance flooding vs. the path-based heuristic/no flooding method suggest that the path-based notion of importance was relatively important in our results. Finally, although not surprisingly, the acceptance of Hypotheses H5a and H5b demonstrate that the importance flooding approach achieved near-perfect results when perfect importance rules were available.

Table 4.2. Significance Test Results

Techniques: • BFS = breadth first (rank by # of hops) • CA = closest associates • IMP = importance flooding • PATH = path heuristics, no flooding • NO = only node heuristics, flooding • PIF = perfect information flooding
All techniques improve on BFS • H1a: A(IMP) < A(BFS) Accepted • H1b: A(CA) < A(BFS) Accepted
Importance flooding outperforms closest associates • H2: A(IMP) < A(CA) Accepted
Importance flooding outperforms heuristics with no flooding • H3: A(IMP) < A(PATH) Accepted at 500,1000 & 2000 but NOT for 100,250
Path heuristics improve on node only heuristics • H4: A(IMP) < A(NO) Accepted
Given "Perfect" information outperforms other techniques • H5a: A(PIF) < A(IMP) Accepted • H5b: A(PIF) < A(CA) Accepted
These hypotheses are expected to hold when 100, 250, 500, 1000, and 2000 nodes have been selected. Accepted Hypotheses were significant at p=.01 for all levels of selected nodes

Table 4.3. Means and Standard Deviations for Ranking Methods

Avg = Average Number of Nodes Selected Per Correct Node

SD = Standard Devation of Number of Nodes Per Correct Node

In each cell, the top number is the number of correct nodes selected

The second row is the Avg and (SD)

Ranking Methodology:	Perfect	Importance Flooding	Path Importance No Flooding	Node Importance With Flooding	Closest Associate	Breadth First
1 to 100 Nodes Selected	68 1.08 (0.14)	27 3.40 (0.62)	24 3.39 (0.68)	19 3.86 (1.07)	15 4.98 (1.51)	0 N/A
1 to 250 Nodes Selected	69 N/A	48 3.95 (0.71)	49 3.96 (0.69)	42 4.70 (1.00)	31 6.20 (1.43)	23 47.63 (21.62)
1 to 500 Nodes Selected	69 N/A	56 5.53 (1.82)	51 5.76 (2.10)	47 5.76 (2.10)	38 8.68 (2.93)	35 29.68 (23.61)
1 to 1000 Nodes Selected	69 N/A	62 8.99 (3.97)	59 9.58 (4.32)	53 10.92 (5.06)	51 12.47 (4.52)	38 25.06 (17.48)
1 to 2000 Nodes Selected	69 N/A	68 15.71 (7.77)	67 16.37 (7.93)	63 18.23 (8.57)	64 19.41 (7.98)	45 30.04 (13.76)
Until All 69 Correct Nodes Were Selected	101 1.08 (0.14)	2158 16.80 (8.43)	3058 23.69 (12.21)	4407 35.03 (17.88)	4140 33.05 (15.59)	4828 46.92 (17.76)

4.8 Discussion and Future Directions

4.8.1 Discussion

We designed the proposed methodology with the intent of supporting the identification of interesting sub-networks of information in large collections of law enforcement records. The approach aligns well with several important domain-specific considerations. Importance flooding directly employs several domain appropriate notions in its calculations including path-based activity heuristics and activity-oriented notions of association closeness. Inquiry specific information can be leveraged by the algorithm. For example, if two individuals are boyfriend/girlfriend it may not be reflected in the data.

This information can be used directly by the importance flooding algorithm simply by adding a few new relations into the set of associations used in the process. Heuristics can be tuned to a particular investigation. In this case, fraud-drug-assault was the focus. The algorithm is also target focused. The algorithm is reasonably tolerant of missing data and can employ ambiguously represented associations. Seemingly unimportant nodes can be drawn into the network when they create a bridge to apparently important nodes.

4.8.2 Conclusions

Our analysis shows the algorithm's promise. In our experimental results we see that the intelligent methods generally outperformed breadth first search. Adding importance heuristics to the analysis seems to improve on an approach which uses only association closeness analysis. Our somewhat novel use of path-based heuristics improved on results achieved using only node-based heuristics. The "perfect" example shows that, given perfect input, importance flooding approaches ideal accuracy. This suggests that inclusion of more and more accurate heuristics or more accurate information can improve on the accuracy achieved using the relatively simple rules and data properties used in this study.

Still, we should be cautious. Because of restrictions on the sharing of information about old investigations, we only tested on two link charts. And even then, the nodes included in the manually prepared link chart are a "bronze standard" rather than a "gold standard" because it may be that some individuals "should" have been included but were not because they were missed by the analyst or left off for a variety of reasons. If an individual was in prison or working with the police as an informant, they may have been

omitted from the chart. Thus we have no real objective standard to say that one chart is "correct" while all others are "incorrect." Instead we would argue that some charts are clearly better than others.

The methodology proposed here allows the processing of large, reasonably-available data sets. The experiments we ran employed only data that can be easily extracted from police records. Only the most basic data items were used; we did not, for instance, employ any crime details beyond a high-level classification of crime type, crime date, and individual's roles in the crime. Yet, the design of the algorithm allows for inclusion of additional "annotation" information for nodes. Nodes could be assigned to groups based on, for instance, their association with border crossing vehicles. Thus the algorithm is appropriately architected for a cross-jurisdictional data-sharing environment.

4.8.3 Future Directions

More work can certainly be done in the law enforcement domain. We would like to study test cases more deeply to address several practical questions. Are the nodes we "suggest" good ones for analysis but left off the charts for a specific reason? How much can we improve results by adding query specific data to the importance ranking calculations? Is the technique useful for creating link charts with various purposes? Does inclusion of locations, vehicles, and border crossings enhance analysis? We plan to implement some version of the algorithm in the AI Lab visualizer to support this kind of detailed work.

In addition, we plan to test importance flooding in other informal node-link knowledge representations. This algorithm is designed to overcome link and identifier

ambiguity by leveraging both the structure and the semantics of the underlying network. The technique presented here allows us to test this basic notion in other application domains. For example, we plan to explore the use of this algorithm in selecting interesting subsets of a network of biomedical pathway relations extracted from the text of journal abstracts.

4.9 Acknowledgements

This work was supported in part by the NSF, Knowledge Discovery and Dissemination (KDD) # 9983304, June 2003-March 2004, NSF, ITR: "COPLINK Center for Intelligence and Security Informatics Research - A Crime Data Mining Approach to Developing Border Safe Research." Sept. 1, 2003 - Aug. 31, 2004, Department of Homeland Security (DHS) / Corporation for National Research Initiatives (CNRI): "Border Safe," Sept. 2003 - Nov. 2004. We are also grateful to Kathy Martinjak, Tim Petersen, and Chuck Violette for their input.

5 CONTRIBUTIONS AND FUTURE DIRECTIONS

The work presented in this dissertation demonstrates several methods of employing information from existing resources in new ways by representing and processing them as node-link knowledge models. Our experiments highlight the ambiguous nature of these representations but demonstrate that they can still be usefully employed in knowledge management tasks. Taken as a whole, this dissertation explores flexible new methodologies able to accomplish some useful analysis tasks in spite of the characteristic informality that inevitably arises when existing knowledge resources are re-used in support of new analysis tasks. The work can be fruitfully expanded over the next few years by looking more deeply at the proposed techniques in their current domains and by applying the methodologies in new domains.

5.1 Contributions

In the first essay, we enhanced the previously-proposed similarity flooding algorithm with two knowledge-anchoring adaptations. With the adaptations, the similarity flooding approach seemed to work well for human-drawn concept maps. This use of the methodology is somewhat different from the schema-oriented applications considered in previous similarity flooding applications.

In the second essay, we proposed a five-level aggregation framework to organize biomedical pathway relations for visualization and analysis. The framework was reasonably well-received in the bio-computing community and was explored in our implementation of the BioAggregate tagger. Based on our experimental results, we are

continuing to study how automatically-extracted pathway relations can be integrated with existing resources and employed in support of research tasks.

The third essay developed an analysis methodology appropriate for cross-jurisdictional law enforcement data analysis. It includes two key elements: (1) an integration framework which allows for the real restrictions existing in the law enforcement domain and (2) an algorithm for systematically leveraging both association closeness and heuristic importance indicators to support network-based analysis tasks. This analysis methodology can be used to guide cost-effective cross-jurisdictional data sharing efforts that respect the many restrictions placed on law enforcement information while also promising systematic and effective support for investigations. Effectiveness results from the integrated use of both semantic elements (importance heuristics) and network structure (incident-based association closeness).

5.2 Informal Knowledge Networks for Decision Support

The essays presented in Chapters 2, 3, and 4 all explore methods of retrieving information, from informal concept graphs, in support of specific analysis processes intended to support decision making. In each case, we show that key organizational tasks can be addressed using algorithms tuned to work despite the informal nature of the underlying data representations.

In our concept mapping application (Chapter 2), we aim to facilitate effective communication of knowledge from instructor to student. Or we might say we want to facilitate knowledge acquisition by students in an instructional setting. Hopefully we can help an instructor answer some important questions such as “What suggestions would

help this particular student?" or "What concepts should be emphasized in upcoming lectures?" Thus our methodology attempts to go beyond simply assigning a score to a concept map (done previously by largely structural matching) and allows map structure to help us overcome inevitable terminology variation that will otherwise muddle the results of a term-only-based match.

In the biomedical application (Chapter 3) we intend to help researchers identify potentially useful areas of investigation. In some sense, more research is needed only when carefully coded data captured in an appropriately specific representation is not available. While there is certainly a place for precise pathway data accessible to precise query processing, there is also a need for investigational support that goes beyond keyword search even when precisely coded relations are unavailable.

The law enforcement application (Chapter 4) also emphasizes the need to combine the weighted associational structure used in previous research with a notion of importance in supporting investigational processes. If we are to efficiently develop understandable, actionable intelligence to support cross-jurisdictional investigations we need more than the simple indexed search available to most agencies and we need to leverage more heuristic knowledge than can be captured by association weight structures alone.

The concept mapping and link chart applications emphasize the need to leverage both structure and semantic meaning in processing this kind of network while the biomedical application establishes a methodology so that more useful processing can be accomplished. We observe that methodologies combining structure and semantics result in somewhat in-elegant solutions. Systems that can rely on strong semantics (description

logic inference tools, for example) might produce more predictable and more definable results. Unfortunately, available data in the domains studied here (and many other domains) cannot be easily or automatically converted into appropriately exacting representations. And even when some particular high-value application justifies creation of extensive and precise (formal) data representations, we observe that such a representation will be quite informal relative to a new line of inquiry based on a slightly different organizational paradigm. Still, this work demonstrates that there is practical value in the relational structure of available information even when available structural information is informally represented. Thus, while we acknowledge that the kind of algorithms presented here may be strongly affected by differences in network structure, subject to subjective measurement of link and node feature values or hindered by inconsistently-expressed heuristics, we believe that more study can be of great benefit in supporting a wide variety of organizational decision-making processes.

5.3 Expanding in Previous Topical Domains

A standalone version of the GetSmart concept mapping tool has already been created. It enables meta-search using the terms found in a concept map and can be easily adjusted to employ a two-tier link structure allowing for both the flexibility required by human users and some additional link categorization to provide a little more data useful in concept matching. These capabilities can be tested in a classroom setting to explore three research questions. (1) Can we improve knowledge acquisition processes involving concept maps by adding search and link categorization features to a concept mapping tool? (2) Can we more effectively match student and master concept maps when they

include categorized links? (3) How can collections of categorized, annotated concept maps be usefully employed in a digital library setting? Positive results connected with any of these questions have potential to contribute to our understanding of computer supported educational processes and to semantic network applications in general.

Node-link representations of the vast available collections of biomedical information promise to support a variety of research endeavors in the coming years, but we need to improve on visualization and analysis application effectiveness. In current and future work, we can explore several important research questions including: Are network representations of automatically-extracted biomedical pathway relations useful in supporting research tasks? Are the relations aggregated by the BioAggregate methodology actually considered equivalent by human users over the aggregation levels described in our framework? How can automatically extracted and aggregated relations be effectively combined with other resources to support research tasks? These three questions can be addressed in the near term (some of the work is already underway). In addition, the importance ranking methodology from Section 4 may be quite useful in helping researchers identify interesting sub-networks of relations in a combined collection of manually-curated and automatically-extracted biomedical pathway relations.

It is very important that effective data sharing methodologies for law enforcement be developed. It seems clear that a lot of data is available to support investigations related to local and national security and law enforcement concerns but sharing that data is very difficult for a variety of reasons. The work presented in Section 4 is promising but it needs to be better validated to have a real impact on investigational practices. Several

immediately testable research questions quickly come to mind. Do the results shown here hold in other investigational settings? Does the addition of additional information (notably location data and border-crossing annotations) improve the effectiveness of the network-ranking technique? Are analysts able to effectively express their intuitions as path-based heuristics in a variety of investigational settings? The answers to these and other related questions should add to the credibility of systematic criminal activity network analysis as an implementable, cross-jurisdictional investigation methodology.

5.4 Expanding in New Topical Domains

In addition to exploring the research questions listed in Section 5.3, I expect that new domains can also be explored. The techniques presented here are potentially applicable for a variety of applications. These applications are characterized by:

1. a strong relational component,
2. a domain-appropriate network representation, and
3. pronounced informality in the network.

If flat structures (e.g., lists), specific selection criteria, and unambiguous feature assignments are appropriate for decision-supporting analysis tasks (as in "list the average age of recent Ph.D. candidates who have successfully completed their dissertations), techniques like those presented in this dissertation are not necessary. On the other hand, when relevant information has been carefully pre-processed to prepare for specific analysis processes (as in expert system representations) the class of techniques we discuss will likely not be as accurate as algorithms based on more formal logic and structure. Still there are many decision making domains which could benefit from analysis techniques

able to leverage informal networks representations of knowledge. The most promising of these applications probably relate to analysis tasks undertaken by researchers, analysts, investigators, and knowledge workers (like instructors) who need to choose between a constantly changing range of possible actions based on ambiguous data and evolving insights.

5.5 Relevance to Business and Managed Organizations

Effectively managing available knowledge resources is vital for today's businesses and organizations. The tremendous network technology advances and investments made in the last two decades easily enable a superficial sharing of vast quantities of potentially useful data and information. However, that information is collected and organized using different paradigms with different tasks in mind. Extensive research and investment in ontologies, relevance-based search techniques, and data integration methodologies allow for some successful examples of information sharing. But analysis techniques able to enhance the usefulness of existing resources in new and multi-source tasks still promise to deliver value in a variety of knowledge acquisition and management tasks.

Business intelligence gathering lends itself to informal knowledge network processing. It is clear that successfully scanning the environment for opportunities and threats is vital to organizational success in today's rapidly changing and global environment. Vast heterogeneous knowledge resources are available internally, from the internet, and from specialty information providers, but organizing, filtering, and merging those resources is difficult as formats change, new resources become available, and new

insights color the scanning process. A variety of techniques have been developed to automatically extract relational information from free text documents. For example, readily available web pages contain key bits of business intelligence but those bits can be obscured by the vast amount of data available on the Internet. NLP processes are increasingly able to identify entities in text and even extract relations between entities but organizing the extracted information into a useful network is still a difficult and under-studied task. While a small portion of the useful relations contained in business documents can be extracted and stored in precise relational representations, we expect a larger portion can be captured in somewhat ambiguous models. These models may allow more intense analysis of available information than what is possible using only keyword retrieval but are not likely to support the kind of precise logical inference associated with many artificial intelligence techniques. The techniques described here promise to support some useful analysis processes that fall between the top-down, model-rich approaches used in expert systems and the bottom-up, term-rich processes used in document retrieval.

Businesses today operate as a web of relationships connecting people, suppliers, customers, and other stakeholders and the relationships between those entities are complex and ever changing. Existing taxonomies of entities and features and existing lexicons of product and organization names can help us identify important entities in web pages and reports but these resources are inevitably incomplete and often it is relational information that is most important. For example, regulators might want to analyze the connection between important corporate officers as they relate to multiple organizations.

If a particular board member is found to be involved with more than one competing organization or with a chain of suppliers and customers, some investigation may be warranted. Relevant information may be readily available but difficult to process.

The techniques explored in this dissertation can be combined to help analyze networks of business relationships. A system could identify and classify (identify features for) the people and organizations represented in a set of documents to create a set of small semantic maps. The maps could then be aligned by matching elements and organized to create a more concise network of relations. Importance flooding could then be used to select a manageable subset of relations leveraging both automatically-extracted and systematically recorded information. Informal network methodologies are needed because new companies pop up daily, relationships are difficult to assess using only available texts, and the names of key entities are ambiguously represented. (1) One name string can be used to represent many different entities (e.g., Bob Smith.) (2) Several name strings can be used for the same entity (e.g., MFST & Microsoft.) (3) Entities are expressed over multiple levels of granularity (e.g., IBM consulting vs. IBM.) While extensive research has been done to normalize selected entities in the business domain, the informal network approaches presented here may help in extracting relational information of use in business intelligence applications.

In addition to intelligence gathering, informal network techniques are potentially useful in supporting operational activities. For example, the informality seen in human-drawn concept maps is somewhat like the ambiguity one would expect to find in collections of workflow models and web service descriptions. The usefulness of

similarity flooding for workflow models is described in (Madhusudan, Zhao, & Marshall, 2004). These models, like concept maps, associate elements as a network. The elements may have different properties and matching elements in different models may be expressed with different granularities, different cardinalities, and ambiguously similar association types. The basic notion of connected modules or functions is likely to hold but different organizations or different designers are likely to use overlapping but variant names for the described processes and features. Establishing mappings between such process models is important if existing process modules are to be re-used but may require a substantial human effort. This cost might be substantially reduced using the knowledge-anchoring similarity flooding technique described in this work.

Research and development is subject to a number of intellectual property issues. Globalization and the diffusion of information via the internet make it increasingly important for organizations to analyze and understand the work being done by others both to recognize new discoveries and to protect internally developed ideas. To support this important agenda, many organizations want to discover new and interesting relations from rich information sources which are more structured than free text but require ad-hoc merging of items from heterogeneous sources. For example, biotech companies carefully monitor patent applications, annotations to the publicly-available GO Ontology, proprietary gene databases, and news articles hoping to discover new opportunities for research and development and tracking the application of intellectual property. New questions and new insights frequently arise as highly trained researchers perform this kind of environmental scanning. Flexible analysis algorithms able to quickly re-analyze

previous data, while including both new heuristics and new previously unrecorded relationships, may be useful in supporting this class of integrated, human-guided analysis. The algorithms presented here, while they do not claim to have the analytic power of deductive reasoning systems, promise to help people analyze and leverage information from existing, heterogeneous and informal knowledge resources. This capability is important as organizations attempt to collaboratively and systematically construct knowledge from the ever-increasing flood of available information.

6 REFERENCES

Ackoff, R. L. (1989). From Data to Wisdom. *Journal of Applied Systems Analysis, 16*, 3-9.

Alavi, M., & Leidner, D. E. (2001). Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. *MIS Quarterly, 25*(1), 107-136.

All, A. C., & Havens, R. L. (1997). Cognitive/Concept Mapping: A Teaching Strategy for Nursing. *Journal of Advanced Nursing, 25*(6), 1210-1219.

ARJIS.*Automated Regional Justice Information System*. Retrieved 3/24, 2004, from the World Wide Web: http://www.arjis.org/

Atabakhsh, H., Larson, C., Petersen, T., Violette, C., & Chen, H. (2004). *Information Sharing and Collaboration Policies Within Government Agencies.* Paper presented at the 2nd Symposium on Intelligence and Security Informatics, June 10-11 2004, Tucson, AZ.

Ault, C. R. (1985). Concept Mapping as a Study Strategy in Earth Science. *Journal of Science College Teaching*, 38-44.

Ausubel, D. P. (1968). *Educational Psychology: A Cognitive View*. New York: Rinehart and Winston.

Bayse, W. A., & Morris, C. G. (1987). FBI Automation Strategy: Developing AI Applications for National Investigative Programs. *Signal Magazine, May*, 185-203.

Berners-Lee, T., Connolly, D., & Swick, R. R. (1999). *Web Architecture: Describing and Exchanging Data*. W3C Note 7. Retrieved April 9, 2005, from the World Wide Web: http://www.w3.org/1999/04/WebData

Brachman, R. J., McGuinness, D. L., Patel-Schneider, P. F., & Borgida, A. (1999). Reducing CLASSIC to Practive: Knowledge Representation Theory Meets Reality. *Artificial Intelligence, 114*, 203-237.

Canas, A. J., Coffey, J. W., Reichherzer, T., Hill, G., Suri, N., Carff, R., Mitrovich, T., & Eberle, D. (1998). *A Performance Support System with Embedded Training for Electronics Technicians.* Paper presented at the The Proceedings of the Eleventh Annual Florida Artificial Intelligence Research Society Conference.

Canas, A. J., Leake, D. B., & Maguitman. (2001). *Combining Concept Mapping with CBR: Experience-based Support for Knowledge Modeling.* Paper presented at the Fourteenth International Florida Artificial Intelligence Research Society Conference.

Canas, A. J., Leake, D. B., & Wilson, D. C. (1999). *Managing, Mapping and Manipulating Conceptual Knowledge: Exploring the Synergies of Knowledge Management & Case-based Reasoning.* Paper presented at the AAAI Workshop on Exploring Synergies of Knowledge Management and Case-based Reasoning, Menlo Park, CA.

Carnot, M. J., Dunn, B., Canas, A. J., Gram, P., & Muldoon, J. (2001). *Concept Maps vs. Web Pages for Information Searching and Browsing*. Institute for Human and Machine Cognition. Retrieved 7/29/2002, 2002, from the World Wide Web: http://www.coginst.uwf.edu/~acanas/Publications/CMapsVSWebPagesExp1/CMapsVSWebPagesExp1.htm

Chabrow, E. (2002, January 14th). Tracking The Terrorists: Investigative Skills and Technology are Being Used to Hunt Terrorism's Supporters. *Information Week*.

Chen, H. (2001). *Knowledge Management Systems - A Text Mining Perspective*: University of Arizona, Managment Information Systems Department.

Chen, H., Schroeder, J., Hauck, R. V., Ridgeway, L., Atabakhsh, H., Gupta, H., Boarman, C., Rasmussen, K., & Clements, A. W. (2002). COPLINK Connect: Information and Knowledge Management for Law Enforcement. *Decision Support Systems, 34*, 271-285.

Chen, H., Zeng, D. D., Atabakhsh, H., Wyzga, W., & Schroeder, J. (2003). COPLINK Managing Law Enforcement Data and Knowledge. *Communications of the ACM, 46*(1), 28-34.

Chen, I.-M. A., & Rotem, D. (1998). *Integrating Information from Multiple Independently Developed Data Sources.* Paper presented at the 7th International Conference on Information and Knowledge Management, Bethesda, Maryland.

Chen, S.-W., Lin, S. C., & Chang, K. E. (2001). Attributed Concept Maps: Fuzzy Integration and Fuzzy Matching. *IEEE Transactions on Systems, Man, and Cybernetics, 31*(5), 842-852.

Chmielewski, T. L., & Dansereau, D. F. (1998). Enhancing the Recall of Text: Knowledge Mapping Training Promotes Implicit Transfer. *Journal of Educational Psychology, 90*(3), 407-413.

Chmielewski, T. L., Dansereau, D. F., & Moreland, J. L. (1997). Updating Knowledge: The Roles of Format, Editing, and Information Type. Presented at the annual meeting of the Midwestern Psychological Association.

Cleveland, H. (1982). Information As a Resource. *The Futurist*, 34-39.

Coady, W. F. (1985). Automated Link Analysis - Artificial Intelligence-Based Tool for Investigators. *Police Chief, 52*(9), 22-23.

Coffman, T., Greenblatt, S., & Marcus, S. (2004). Graph-Based Technologies for Intelligence Analysis. *Communications of the ACM, 47*(3), 45-47.

Dalgarno, B. (2001). Interpretations of constructivism and consequences for computer assisted learning. *British Journal of Educational Technology, 32*(2), 183-194.

Davenport, T. H., & Prusak, L. (2000). *Working Knowledge: How Organizations Manage What They Know*. Boston, MA: Harvard Business School Press.

Doan, A., Lu, Y., Lee, Y., & Han, J. (2003). Profile-Based Object Matching for Information Integration. *IEEE Intelligent Systems, 18*(5), 54-59.

Eliot, T. S. (1934). *The Rock* (First ed.). New York, New York: Harcourt, Brace and Company.

Evans, S. H., & Dansereau, D. F. (1991). Knowledge Maps as Tools for Thinking and Communication. In R. F. R. F. Mulcahy & R. H. Short & J. Andrews (Eds.), *Enhancing learning and thinking* (pp. 291). New York: Praeger.

Ford, K., Canas, A. J., & al., e. (1995). *Knowledge Construction and Sharing in Quorum.* Paper presented at the World Conferenece on Artificial Intelligence in Eduation.

Ford, K., Coffey, J. W., Canas, A. J., Andrews, E. J., & Turner, C. W. (1996). Diagnosis and Explanation by a Nuclear Cardiology Expert System. *International Journal of Expert Systems, 9*, 499-506.

Friedman, C., Kra, P., Yu, H., Krauthammer, M., & Rzhetsky, A. (2001). GENIES: a Natural-Language Processing System for the Extraction of Molecular Pathways from Journal Articles. *Bioinformatics, 17*, S74-82.

Fukuda, K., Tsunoda, T., Tamura, A., & Takagi, T. (1998). *Toward Information Extraction: Identifying Protein Names from Biological Papers.* Paper presented at the Pac. Symp. Biocomput., Big Island, Hawaii.

Furnas, G. W., Landauer, T. K., Gomez, L. M., & Dumais, S. T. (1987). The Vocabulary Problem in Human-System Communication. *Communications of the ACM, 30*(11), 964--971.

Gaines, B. R. (1995). Class Library Implementation of an Open Architecture Knowledge Support System. *International Journal of Human-Computer Studies, 41*(1-2), 59-107.

Gaines, B. R., & Shaw, M. L. G. (1995). Concept Maps as Hypermedia Components. *International Journal of Human-Computer Studies, 43*(3), 323-361.

Gaines, B. R., & Shaw, M. L. G. (1995). *WebMap: Concept Mapping on the Web.* Paper presented at the Proceedings of WWW4: Fourth International World Wide Web Conference, Boston.

Gaizauskas, R., Demetriou, G., Artymiuk, P., & Willett, P. (2003). Protein structures and information extraction from biological texts: the PASTA system. *Bioinformatics, 19*(1), 135-143.

Garcia-Molina, H., & Ullman, J. D. (2002). *Database Systems: The Complete Book*. New Jersey: Prentice Hall.

Gehrke, J., Ginsparg, P., & Ginsparg, P. (2003). Overview of the 2003 KDD Cup. *SIGKDD Explor. Newsl., 5*(2), 149-151.

Hall, R. H., & Odonnell, A. (1996). Cognitive and Affective Outcomes of Learning from Knowledge Maps. *Contemporary Educational Psychology, 21*(1), 94-101.

Hanisch, D., Fluck, J., Mevissen, H., & Zimmer, R. (2003, January 3-7, 2003). *Playing Biology's Name Game: Identifying Protein Names in Scientific Text.* Paper presented at the Pac. Symp. Biocomput., Lihue, Hawaii, USA.

Harper, W. R., & Harris, D. H. (1975). The Application of Link Analysis to Police Intelligence. *Human Factors, 17*(2), 157-164.

Hartshorne, C., Weiss, P., & Burks, A. (Eds.). (1931-1935). *The Collected Papers of Charles Sanders Peirce* (Vol. 4). Cambridge, MA: Harvard University Press.

Herl, H. E., & al., e. (1999). Reliability and Validity of a Computer-Based Knowledge Mapping System to Measure Content Understanding. *Computers in Human Behavior*(15).

Hilderman, R. J., & Hamilton, H. J. (2001). Evaluation of Interestingness Measures for Ranking Discovered Knowledge. *Lecture Notes in Computer Science, 2035*, 247-259.

Hirschman, L., Morgan, A. A., & Yeh, A. S. (2002). Rutabaga by Any Other Name: Extracting Biological Names. *J. Biomed. Inform., 35*, 247-259.

Hoffman, R. R., Coffey, J. W., Ford, K., & Carnot, M. J. (2001). *Storm-1k: A Human-Centered Knowledge Model for Weather Forecasting.* Paper presented at the The Proceedings of the 45th Annual Meeting of the Human Factors and Ergonomics Society, Minneapolis, MN.

I2. (2004). *I2 Investigative Analysis Software*. Retrieved November 29, 2004, from the World Wide Web: http://www.i2inc.com/Products/Analysts_Notebook/#

Kaza, S., Xu, J., Marshall, B., & Chen, H. (2005). Topological Analysis of Criminal Activity Networks: Enhancing Transportation Security. *IEEE Transactions on Intelligent Transportation Systems, Under Review*.

KCC. (2004). *COPLINK from Knowledge Computing Corp.* Retrieved November 29, 2004, from the World Wide Web: http://www.coplink.net/vis1.htm

Kinchin, I. M. (2001). If Concept Mapping Is So Helpful to Learning Biology, Why Aren't We All Doing It? *International Journal of Science Education, 23*(12), 1257-1269.

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

Krebs, V. E. (2001). Mapping Networks of Terrorist Cells. *Connections, 24*(3), 43-52.

Kremer, R. (1994). *Concept Mapping: Informal to Formal.* Paper presented at the Proceedings of the International Conference on Conceptual Structures, University of Maryland.

Lambiotte, J. G., Dansereau, D. F., Cross, D. R., & Reynolds, S. B. (1989). Multirelational Semantic Maps. *Educational Psychology Review, 1*, 331-367.

Lanckriet, G. R. G., Deng, M., Cristianini, N., Jordan, M. I., & Noble, W. S. (2004, January 6-10, 2004). *Kernel-based Data Fusion and its Application to Protein Function Prediction in Yeast.* Paper presented at the Pac. Symp. Biocomput., Big Island, Hawaii.

Lawless, C., Smee, P., & O'Shea, T. (1998). Using Concept Sorting and Concept Mapping in Business and Public Administration, and in Education: an Overview. *Educational Research, 40*(2), 219-235.

Leake, D. B., Maguitman, A., & Canas, A. J. (2002). *Assessing Conceptual Similarity to Support Concept Mapping.* Paper presented at the Proceedings of FLAIRS-02, Pensacola, Florida.

Leake, D. B., Maguitman, A., Reichherzer, T., Canas, A. J., Carvalho, M., Arguedas, M., Brenes, S., & Eskridge, T. (2003). *Aiding Knowledge Capture by Searching for Extensions of Knowledge Models.* Paper presented at the To appear in K-Cap-03, Sanibel Island, Florida.

Leroy, G., Chen, H., & Martinez, J. D. (2003). A Shallow Parser Based on Closed-Class Words to Capture Relations in Biomedical Text. *J. Biomed. Inform., 36*, 145-158.

Lim, E.-P., Srivastava, J., Prabhakar, S., & Richardson, J. (1996). Entity Identification in Database Integration. *Information Sciences, 89*(1), 1-38.

Lin, S.-d., & Chalupsky, H. (2003). Using Unsupervised Link Discovery Methods to Find Interesting Facts and Connections in a Bibliography Dataset. *SIGKDD Explor. Newsl., 5*(2), 173-178.

Madhusudan, T., Zhao, J. L., & Marshall, B. (2004). A Case-Based Reasoning Framework for Workflow Model Management. *Data & Knowledge Engineering, 50*(1), 87-115.

Marshall, B., Kaza, S., Xu, J., Atabakhsh, H., Petersen, T., Violette, C., & Chen, H. (2004). *Cross-Jurisdictional Criminal Activity Networks to Support Border and Transportation Security.* Paper presented at the 7th International IEEE Conference on Intelligent Transportation Systems, Washington D.C.

Marshall, B., Su, H., McDonald, D. M., & Chen, H. (2005). *Linking ontological resources using aggregatable substance identifiers to organize extracted relations.* Paper presented at the Pac. Symp. Biocomput., Big Island, Hawaii.

Marshall, B., Zhang, Y., Chen, H., Lally, A., Shen, R., Fox, E., & Cassel, L. (2003). *Convergence of Knowledge Management and E-Learning: the GetSmart Experience.* Paper presented at the Joint Conference on Digital Libraries, Houston, Texas.

McDonald, D. M., Chen, H., Su, H., & Marshall, B. B. (2004). Extracting Gene Pathway Relations using a Hybrid Grammar: the Arizona Relation Parser. *Bioinformatics, 20*(18), 3370-3378.

Melnik, S., Garcia-Molina, H., & Rahm, E. (2001). *Similarity Flooding: a Versatile Graph Matching Algorithm (Extended Technical Report)*. Retrieved, 2003, from the World Wide Web: http://dbpubs.stanford.edu:8090/pub/2001-25

Melnik, S., Garcia-Molina, H., & Rahm, E. (2002). *Similarity Flooding: a Versatile Graph Matching Algorithm and its Application to Schema Matching.* Paper presented at the Proceedings of the 18th International Conference on Data Engineering (ICDE '02), San Jose, Ca.

Nosek, J. T., & Roth, I. (1990). A Comparison of Formal Knowledge Representation Schemes as Communication Tools: Predicate Knowledge vs. Semantic Network. *International Journal of Man-Machine Studies, 33*, 227-239.

Novak, J. (1998). *Learning, Creating, and Using Knowledge: Concept Maps as Facilitative Tools in schools and Corporations.* Mahwah, New Jersey: Lawrence Erlbaum Associates.

Novak, J., & Gowin, D. B. (1984). *Learning How To Learn*. Cambridge, UK: Cambridge University Press.

Noy, N. F., & Musen, M. A. (2000). *Prompt: Algorithm and Tool for Automated Ontology Merging and Alignment.* Paper presented at the Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000), Austin, TX, USA (2000).

Ogren, P. V., Cohen, K. B., Acquaah-Mensah, G. K., Eberlein, J., & Hunter, L. (2004, January 6-10, 2004). *The Compositional Structure of Gene Ontology terms.* Paper presented at the Pac. Symp. Biocomput., Big Island, Hawaii, USA.

Padmanabhan, B., & Tuzhilin, A. (1999). Unexpectedness as a Measure of Interestingness in Knowledge Discovery. *Decision Support Systems, 27*(3), 303-318.

Palakal, M., Stephens, M., Mukhopadhyay, S., Raje, R., & Rhodes, S. (2003). Identification of Biological Relationships from Text Documents Using Efficient Computational Methods. *J. Bioinform. Comput. Biol., 1*(2), 307-342.

Park, J. C., Kim, H. S., & Kim, J. J. (2001, January 3-7, 2001). *Bidirectional Incremental Parsing for Automatic Pathway Identification with Combinatory Categorial Grammar.* Paper presented at the Pac. Symp. Biocomp., Hawaii, USA.

Pustejovsky, J., Castano, J., Zhang, J., Kotecki, M., & Cochran, B. (2002, January 3-7, 2002). *Robust Relational Parsing Over Biomedical Literature: Extracting Inhibit Relations.* Paper presented at the Pac. Symp. Biocomput., Lihue, Hawaii.

Rahm, E., & Bernstein, P. A. (2001). A Survey of Approaches to Automatic Schema Matching. *The VLDB Journal, 10*, 334-350.

Rindflesch, T. C., Tanabe, L., Weinstein, J. N., & Hunter, L. (2000). *EDGAR: extraction of drugs, genes and relations from the biomedical literature.* Paper presented at the Pac. Symp. Biocomput., Big Island, Hawaii.

Rye, J. A., & Rubba, P. A. (2002). Scoring Concept Maps: an Expert Map-Based Scheme for Relationships. *School and Science Mathematics, 102*(1), 33--44.

Rzhetsky, A., Iossifov, I., Koike, T., Krauthammer, M., Kra, P., Morris, M., Yu, H., Duboue, P. A., Weng, W., WIlbur, W. J., Hatzivassiloglou, V., & Friedman, C. (2004). GeneWays: a System for Extracting, Analyzing, Visualizing, and Integrating Molecular Pathway Data. *J. Biomed. Inform., 37*, 43-53.

Sahar, S. (2001). *Interestingness Preprocessing.* Paper presented at the Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on.

Sahar, S. (2002). *On Incorporating Subjective Interestingness into the Mining Process.* Paper presented at the Data Mining, 2002. ICDM 2002. Proceedings. 2002 IEEE International Conference on.

Schmitt, R. B. (2005, January 13). New FBI Software May Be Unusable. *Los Angeles Times*.

Schroeder, J., Xu, J., & Chen, H. (2003). *CrimeLink Explorer: Using Domain Knowledge to Facilitate Automated Crime Association Analysis.* Paper presented at the Intelligence and Security Informatics, Proceedings of ISI-2004, Lecture Notes in Computer Science.

Sharma, N. (2005, 2/23/2005). *The Origin of the Data Information Knowledge Wisdom Hierarchy*. University of Michigan, Ann Arbor. Retrieved 4/2/2005, 2005, from the World Wide Web: http://www-personal.si.umich.edu/~nsharma/dikw_origin.htm

Shavelson, R. J., Lang, H., & Lewin, B. (1993). *On Concept Maps as Potential "Authentic Assessments" in Science.* Los Angeles, CA: National Center for Research on Evaluation, Standards, and Student Testing. US Department of Education, Grant R117G10027 (ERIC Document Reproduction Service No. ED 367691).

Silberschatz, A., & Tuzhilin, A. (1996). What Makes Patterns Interesting in Knowledge Discovery Systems. *IEEE Transactions on Data and Knowledge Engineering, 8*, 970-974.

Sowa, J. F. (2000). *Knowledge Representation: Logical, Philosophical, and Computational Foundations*: Brooks/Cole.

Sparrow, M. K. (1991). The Application of Network Analysis to Criminal Intelligence: An Assessment of the Prospects. *Social Networks, 13*(3), 251-274.

Staff. (2003, November 23). Software Joins Cops on the Beat," COPLINK program links databases, speeds police investigations in the state of Alaska. *Anchorage Daily News*.

Tuason, O., Chen, L., Liu, H., Blake, J. A., & Friedman, C. (2004, January 6-10, 2004). *Biological Nomenclatures: a Source of Lexical Knowledge and Ambiguity.* Paper presented at the Pac. Symp. Biocomp., Big Island, Hawaii.

USDJ. (2004). *Office of Justice Programs, global justice XML data model*. US Department of Justice. Retrieved February 2, 2004, from the World Wide Web: http://www.it.ojp.gov/topic.jsp?topic_id=43

Wang, G., Chen, H., & Atabakhsh, H. (2004). Automatically Detecting Deceptive Criminal Identities. *Communications of the ACM, 47*(3), 70-76.

Weideman, M., & Kritzinger, W. (2003). *Concept Mapping vs. Web Page Hyperlinks as an Information Retrieval Interface: Preferences of Postgraduate Culturally Diverse Learners.* Paper presented at the Proceedings of the 2003 annual research conference of the South African institute of computer scientists and information technologists on enablement through technology.

White, S., & Smyth, P. (2003). *Algorithms for Estimating Relative Importance in Networks.* Paper presented at the ACM SIGKDD internt'l conference on knowledge discovery and data mining, Washington, D. C.

Xu, J., & Chen, H. (2003). *Untangling Criminal Networks: A Case Study.* Paper presented at the NSF/NIJ Symposium on Intelligence and Security Informatics (ISI), Tucson, AZ.

Xu, J., & Chen, H. (2004). Fighting Organized Crime: Using Shortest-Path Algorithms to Identify Associations in Criminal Networks. *Decision Support Systems, 38*(3), 473-487.

Yeh, A. S., Hirschman, L., Morris, M., & Colosimo, M. ((2004)). *BioCreAtIve task 1A: gene mention finding evaluation*. Bedford, MA: The MITRE Corporation.

Zeleny, M. (1987). Management Support Systems: Towards Integrated Knowledge Management. *Human Systems Management, 7*(1), 59-70.