A DATA MINING APPROACH
TO IDENTIFY PERPETRATORS:
AN INTEGRATION FRAMEWORK
AND CASE STUDIES

by

LI DING

A DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Computer Science
in the Graduate School of
The University of Alabama

TUSCALOOSA, ALABAMA

2010

ABSTRACT

Data mining and social network analysis have been widely used in law enforcement to solve crimes. Research questions such as strength of ties in social networks, crime pattern discovery and prioritizing offenders have been studied in this area. However, most of those studies failed to consider the noisy nature of the data. The techniques they proposed only have been applied to small scale data sets. Therefore, it is an important task to design a framework that can work on large scale data sets and tolerance noisy data.

In this dissertation, we built an integrated crime detection framework that combined two data mining techniques: decision tree and genetic algorithm and graph theories to solve the problems we pointed out. Our crime pattern analysis is based on all offenders of the state of Alabama in the past 50 years. Our constructed social network contains all Alabama residents. It allows us to fully evaluate the proposed models.

Two case studies have been conducted to evaluate the framework. One is based on 625 inmates released from Madison county jail in 2004. Our experimental results show that our recommended risk level has strong correlation in predicting future offense. Another case study is based on the 100 real police reports. The experimental results show that the median ranking of arrestees remains at the top 3% of the return list.

LIST OF ABBREVIATIONS AND SYMBOLS

AUC	Area under the receiver operator characteristic. Strength of the association of two random variables, one of which is a binary variable.
SNA	Social network analysis
G	A graph
V	Vertices set of a graph
E	Edge set of a graph
O	An upper bound of the complexity of a function
e	an edge in a graph
v	a vertex in a graph
p	Probability associated with the occurrence under the null hypothesis of a value as extreme as or more extreme than the observed value

ACKNOWLEDGMENTS

First of all, I would like to thank my advisor, Dr. Brandon Dixon, for giving me a chance to join this program and providing guidance, advice and encouragement. Thank you for sharing your research wisdom with me and kindly help me on research problems.

I would also like to acknowledge my committee members Dr. Allen Parrish, Dr. Randy Smith, Dr. Xiaoyan Hong, and Dr. David Forde for their valuable suggestions on my dissertations and academic progresses.

This research would not have been possible without the support from colleagues from CAPS (Center of Advanced Public Safety) at UA, my family and friends.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In the past decade, electronic systems used in law enforcement departments have created rich data sources for investigators and researchers to analyze. Several applications have been built in recent years to aid the investigation by using data mining methods [1, 2], social network analysis [3], or geographical profiling [4-6]. Another area that draws a lot of attention from researchers and correction officers is inmate risk assessment systems which are mainly based on the analysis of criminal history [7-12].

In crime investigation area, researchers [1, 2] have applied clustering technique to find crime patterns of a certain type of crimes; outlier detection technique to indentify abnormal activities; classification technique to find similar properties among criminal history and organized them to different categories. These methods have been tested on small scale data sets. They assume high data quality and fail to consider the noisy nature of the data sources. Researchers [4-6] from criminal justice domain have proposed using geographical profiling to identify the most likely locations the suspect lived in. This technique is based on the geographical analysis of the crime scene location and assumption of how the suspect travels to crime scene. However, this approach only answers "where" the suspect is and can't answer the question "who" commits the crime.

Social network analysis [3] has also been studied in organized crime investigation. Studies are mainly based on how to discover relationships in a given criminal network and how close any given two people are in the discovered social networks. The goal of the studies is to construct the social network close to real world criminal organization. Therefore, it's important

to have a measurement to measure how strong the relationship is of any two given people in the network. Previous studies have suggest using frequency of co-occurrence of two entities as the strength of two directly connected people and shortest path algorithm to measure the strength of two indirectly connected people. However, they all failed to consider the poor quality of raw data. The relationships discovered in the social network construction may not exist in the real world. If the data quality is not good enough, then these approaches won't work well.

Crime patterns associated with repeated offenders combined with other factors can also be used to predict re-offense of offenders. It has drawn great research interests to provide a standard model which has the ability to classify the offenders in terms of the likelihood of their next offenses. There are several systems [7-12] that have claimed that their systems are valid for all types of offenders across all jurisdictions. However, all of the systems need manually input the data by interviewing inmates. It increases the chance of false predicting.

Try to solve these research problems mentioned above, we built a framework that integrates over 50 year's criminal records, driver's license registration information, vehicle registration information, citation information, and correction information of the state of Alabama. We studied two data mining techniques: decision tree and genetic algorithm and graph theories in this framework to assist predicting recidivism, strong tie identification and crime investigation.

In particular, this dissertation contributes to these research areas:

An automatic model to predict recidivism (Chapter 2)

The model we proposed have the following properties: the recommended risk level has a strong correlation to re-offending; the items collected to construct the evaluation of the risk level are accurate and objective. Our research is based on the data collected from the Madison County

Jail in the state of Alabama. This data comprises 20 years of inmates' information and over 50 years of criminal information from multiple law enforcement agencies in Alabama. We decompose the evaluation of risk level into 4 weighted categories which contain a total of 9 weighted items. All of the data can be obtained from a preprocessed dataset to ensure the quality and objectivity of the items. We have used a genetic algorithm to compute the weights associated with each of the categories as well as the individual items so as to optimize the correlation of risk level and recidivism in our sample data. Our experimental results support the two properties we claimed.

A robust measurement to measure strong ties in social networks (Chapter 3)

Granovetter introduced the importance of weak ties in 1973 [13], indicating that often extremely valuable information is dispersed through persons who are not close to each other. However in real-world social networks, especially in criminal networks, people tend to use more "reliable" relations to transmit sensitive information. It seems reasonable that someone who transports drugs would share delivery time information only with those who are "trusted" relations. The most recent study [14] gave two examples where removing weak ties did not destroy the information passing mechanism. Here we give a concrete real example of how strong ties can affect crime investigation. A homicide investigation found that a white Ford truck was seen leaving a homicide scene in Crenshaw County, Alabama in October 2008. Later, after a suspect was identified, it was found that a truck matching this description was registered under the name of the suspect's father. In this example, the father and the son have a strong relationship in the social network, and the suspect would be missed if we fail to consider this strong tie.

Since strong ties play such a crucial role in structuring the social network, identification of the strong ties should be given special consideration. However, SNA must deal with uncertain relations and the noisy data generated by the relation discovery rules and poor quality of the raw data. One challenge when measuring the strength of ties must be for the metrics to be robust towards noise.

In this research, we combine the graph theoretic concepts of *k*-connectivity [15] and the edge-dual graph [16] and propose a novel measure of strength between the ties of any two nodes in the network. To do this, we translate the original social network to a corresponding edge-dual graph by extracting the relation context between connected node pairs and creating a new node for each unique relation context. These new relation nodes are called relation context nodes. Connections are added between the new relation context nodes and their original nodes. After drawing the edge-dual graph we use the *k*-connectivity concept from graph theory to compute the connectivity between two nodes. The *k*-connectivity metric measures how many relation context nodes must be removed to disconnect two given nodes. The most important advantage of doing this is the improvement in the SNA robustness towards noisy data. In particular, a few incorrect relations will not ruin the computation of closeness. Another advantage is that relation context nodes will help users understand the relations between nodes through visualization.

We integrated data from multiple domains into a common data model which is then represented by a social network. We applied statistical analyses to the relationships of 300 co-offenders who have committed robbery together. We found most strong ties have the properties of having *2*-connectivity.

An integrated framework to identify perpetrators (Chapter 4)

To our knowledge, there is no research showing integrated data mining, social network analysis, and geographical profiling tools to aid investigation. We propose an integrated framework called FIRST. FIRST is an inter-disciplinary framework that integrates the before mentioned technologies to aid crime investigators. Given the location of a crime, with or without physical descriptions of suspects (personal characteristics and vehicle descriptions), to solve the crime, FIRST will process the inputs with the following four steps:

1. Apply a geospatial search based on either a default radius which varies by the size of the city or officer selected criteria (such as radius or designated regions).
2. Retrieve all persons who have committed at least one felony crime prior to this crime and have an address inside the search area or have a close relationship determined through our use of social network analysis to an individual with an address inside the search area.
3. Use biometric filtering techniques to reduce the perpetrator search space. Specifically, a fuzzy search returns the persons who fit the physical description within some given tolerances. For example, given the search criteria "a 6 foot male", the search engine will return all the males within 5' 7'' to 6' 3'' to avoid the potential of missing suspects.
4. Use a crime pattern component to rank the qualified suspects based on their criminal history similarities to the crime pattern of current type of crime. Statewide historical crime data is used to analyze the crime patterns for different type of crimes. A modified decision tree model is used to score suspects. For

example, a person with a robbery or burglary charge on his/her record has a higher probability to commit robbery again.

Data sources used in the framework are coming from cross-jurisdictional data such as arrest records, sentencing records, driver's license registration information, vehicle registration information, prison, and jail information etc. We built this framework in a pay-as-you-go fashion which allows other states to adopt this framework easily.

The primary contribution of this work is the introduction of the first (to the best of our knowledge) framework to integrate spatial analysis, social network analysis, crime pattern analysis and biometric matching to assist crime investigation. The flexibility of the framework allows other states or federal agencies to replicate the framework. The decision tree model of crime pattern analysis can be applied to other states with minor modification. Since most states have electronic arrest records, and have similar crime charge codes, the model proposed here can be easily adopted into their systems and have a broader impact. Social network analysis and geographic analysis help officers expand the search beyond simply those individuals who have addresses proximal to the crime.

This dissertation is organized as follows. Chapter 2 describe the model we built to predict recidivism. Chapter 3 is the paper that describes measurement of strong ties in social networks. Chapter 4 is the paper that discusses the FIRST framework. Chapter 5 will has the overall conclusions of this dissertation.

CHAPTER 2

IRAS: AN INMATES' RISK ASSESSMENT SYSTEM

2.1 Introduction

Criminal justice agencies, law enforcement and corrections departments are facing big challenges related to high rates of incarceration in United States. According to a Department of Justice statistics report [18], at the middle of 2007 almost 8 million people are in jail, prison, probation or parole. Several problems emerge when incarceration rates are so high. For example, when a jail is overcrowded, the officer must decide which inmates can be released from jail; a probation manager must allocate the limited resources to the most needed offenders. One of the core components needed to solve these problems is to classify the offenders into different risk levels which is generally called case classification. It has drawn great research interests to provide a standard instrument which has the ability to classify the offenders in terms of the likelihood of their next offenses. To achieve this goal, the instrument or system must have the following properties:

- The recommended risk level must have a strong correlation to re-offending.
- The items collected to construct the evaluation of the risk level must be accurate and objective.

There are several systems that have claimed that their systems are valid for all types of offenders across all jurisdictions. However, other researchers' evaluation of those systems indicates that those systems are potentially weak regarding the two properties listed above. We will discuss the strengths and weaknesses of these systems in detail in the literature review section.

Our system is based on the data collected from the Madison County Jail in the state of Alabama. This data comprises 20 years of inmates' information and over 50 years of criminal information from multiple law enforcement agencies in Alabama. We decompose the evaluation of risk level into 4 weighted categories which contain a total of 9 weighted items. All of the data can be obtained from a preprocessed dataset to ensure the quality and objectivity of the items. We have used a genetic algorithm to compute the weights associated with each of the categories as well as the individual items so as to optimize the correlation of risk level and recidivism in our sample data. Our experimental results show that the risk level we compute has strong correlation to recidivism rate.

The remainder of the paper is organized as follows: we will discuss related work in the Literature Review section; we will describe our system in detail in the System Design section; the experimental design and results is introduced in the Experimental Results section and we will discuss our future work in the Conclusions and Future Works section.

2.2 Literature Review

2.2.1 LSI-R

Over past several years many standardized case classification systems [1, 6, 11, 15] have been constructed and used to study the risk assessment of offenders. Among those systems, LSI-R is the most comprehensive and popular one. LSI-R was developed in the late 1970s in Canada through a collaboration of probation officers, correctional managers, practitioners and researchers. LSI-R is currently being used in a variety of correctional contexts across the United States and has widespread acceptance in North America. LSI-R scores an offender by evaluating 54 items which are believed related to future crime associated with the offender. Those 54 items can be categorized into 10 sub-scales: criminal history, education and employment, financial,

family and marital, accommodations, leisure and recreation, companions, alcohol and drug problems, emotional and personal, and attitudes and orientations. Several researchers found that LSI-R was a valid measure of the likelihood of reoffending [2, 10, 12-14]. However, recent research in Pennsylvania [3] shows that only a few of the factors in LSI-R contribute to the probability of recidivism in felony cases. In other research conducted by Washington State [5], they found that by adding additional factors to supplement LSI-R, they can improve the correlation of LSI-R score and the re-offence likelihood.

One drawback of LSI-R is that the methods used to compute the risk score are static and can't have universal applicability. For instance, the re-offence pattern may be different depending on geographic region or time period. Another issue that LSI-R does not consider is the global optimization of the weighting system of factors. The weights of the factors are assigned based on the analysis of the correlation of only the factors local to that particular sub-score to re-offence rate. When the sub scores are combined together to form the final score of an offender, it may not be the globally optimal solution to predict re-offending.

Motivated by these observations, our system has significant differences from LSI-R. First of all, by taking advantage of current integrated law enforcement data in Alabama, we can build pre-validated models based on historical law enforcement data. Secondly, our models can evolve over time through the periodic use of a genetic algorithm to compute and optimize the weights of the factors.

2.2.2 AUC

The measurement of the strength of the association of risk level and recidivism rate used in our system is called the area under the receiver operator characteristic (AUC) [8]. The traditional statistical method to measure the strength of relationship between two random

variables is called correlation or correlation coefficient. When one of them is dichotomous, researchers use the point biserial correlation coefficient [9] to compute the correlation. However, recent studies have found AUC useful for organizing classifiers and their performance [16]. The AUC ranges from 0.50 to 1.00. This statistic is 0.50 when there is no association and 1.00 when there is perfect association. An AUC above 0.70 indicates a strong association, while measures between 0.60 and 0.70 indicate a moderate association.

2.2.3 Relationship Finder

The relationship finder [7] is a research tool constructed at the University of Alabama. This tool constructs a type of social network where the members of the network are all Alabama residents. The members are connected by a "relation" based on whether they share an address, vehicle, or were involved in a criminal or domestic case, with the data coming from a variety of sources, such as driver and vehicle registrations, citations, and court data. In this paper, we make use of the relationship finder as a data source by examining the "relations" associated with the inmates.

2.3 System Design

We will describe our system in the system architecture, factors and genetic algorithm subsections.

2.3.1 System Architecture

Figure 2.1 shows our system architecture. All of the factors we used to evaluate an offender are computed from attributes stored in one of three different databases: our relationship finder database, the criminal history database or the jail information database. The criminal history database contains 50 years of criminal information from Alabama; the jail information database contains 30 years of inmates booking, release, and occupation information and the

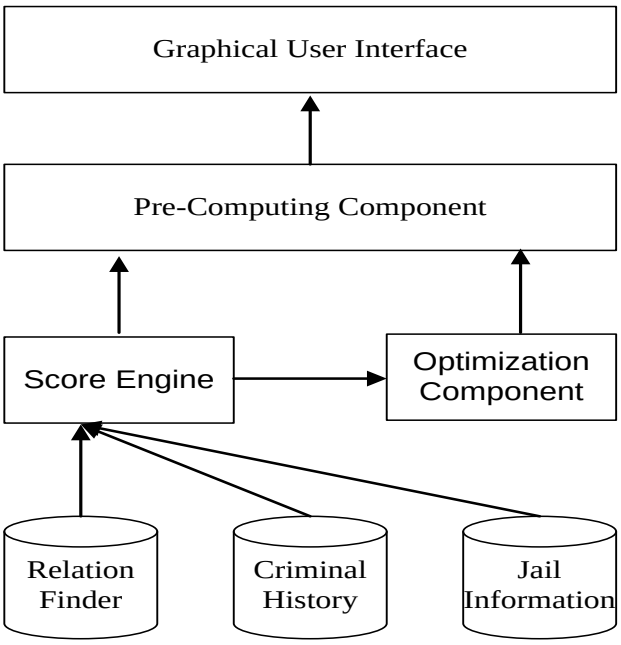


Figure 2.1. System Architecture

relation finder database contains the most recent 5 years of relationships data for all the Alabama residents.

The scoring engine collects each factor from the three databases and scores the factors for each inmate according to the pre-defined equation. This score is then called the raw score in our system.

The optimization component dynamically chooses sample offenders from the database and the raw score of those offenders is computed by the score engine as an input. It then uses a genetic algorithm to evaluate every category and factor so as to optimize the configuration of the weights of the factors to achieve the highest AUC score for the sample data. This optimization process can be initialized by the end-user or as a time based trigger. After computing the best weights for each factor, these weights are given to the pre-computing component for use in computing a final score for the entire set of inmates.

The pre-computing component takes all the current information for the in-jail inmates along with their raw scores from the score engine, applies the weighting function for each factor and calculates the final score for each inmate. After computing the final score, it will give every inmate a risk level recommendation based on their final score. Also, the information from every factor such as a detailed crime description, relations description and occupation will be stored for future retrieval. Since the score is pre-computed, when the inmate is booked or released from jail, the pre-computing component will dynamically update the score.

2.3.2 Factors

The factors used in our system to evaluate offenders are shown in Table 2.1. Since our system is constructed based on the Madison County Jail data, two of the factors (employment before entering jail and age at release) are specific items collected by the jail system itself. However, we expect that the same or similar data is available from other jail systems and therefore factors that are equivalent for our purposes can be created for any other correctional agencies. All the factors have been determined together with expertise from the Madison County

Table 2.1 Factors used to evaluate offenders

Category	Factors	FactorID
Criminal History	Frequency of crime	11
	Highest Severity of crime	12
	Age at first crime	13
	Most recent crime	14
	Number of drug related crimes	15
Relations	How many direct relations have a criminal history	21
	Does the offender have a stable family	22
Education and Employment	Does the offender have a job before entering jail	31
Other Factors	Age at Release	41

Jail system and all of the data can be collected from existing databases. The factors are grouped into 4 categories: criminal history, relations, education and employment, and other factors. Each category has an initial weight and the sum of the four weights is 100. Every factor in each category also has a weight and those weights sum to 100 as well. The score engine evaluates the factors of the inmates and gives each individual factor a score from the range 0 to 100. The computation of the final score of an offender is shown in Equation 1. In Equation 1, variable W_i indicates the weight of category *i*, W'_j indicates the weight of factor *j* and $Score_j$ indicates the score of that factor. *#Categories* represents the number of categories used in the system which in our case is 4. The final score ranges from 0 to 100.

$$Final\ Score = \sum_{i=1}^{\#Categories} \left(W_i \times \sum_{j\ s.t.\ Factor_j \in Category_i} \left(W'_j \times Score_j \right) \right) \quad (1)$$

We have assigned an initial weight to each category and factor based on input from a domain expert. However, as described later, these weights are updated through the use of a genetic algorithm so as to optimize the accuracy of our score based on historical data.

Table 2.2 shows how we score the individual factors. The conditions and scores for each individual factor were again initially created based on consultation with domain experts from the Madison County Jail system. We evaluated the initial set of conditions via statistical analysis, and in some cases, we have been able to simplify the set of conditions for a given factor. For example, consider the categorization of the "Age of first crime" factor shown below. The initial categorization was based on 5 year intervals, but the re-offense pattern for the age range 25 through 40 was not significantly different. We, therefore, combined those categories into the simplified set that you see in the Figure 2.2.

Table 2.2. Description of score engine

FactorID	Condition	Score	Remark
11	*frequency*=0	100	The frequency of offending is counted for all the convicted felony crimes the offender has committed. The crimes also include the data from juvenile court.
	1≤ *frequency* <2	90	
	2≤ *frequency* <4	80	
	4≤ *frequency* <7	40	
	7≤ *frequency* <10	20	
	frequency≥10	0	
12	*minor*	100	The severity level is calculated according the charge the offender has been convicted.
	moderate	80	
	high	40	
13	*age* <20	20	The age of the offender's when he/she committed the first felony crime.
	20≤ *age* <30	50	
	30≤ *age* <40	80	
	age≥40	90	
14	*year* <3	60	How many years since the offender committed last crime?
	3≤ *year* <7	80	
	year≥7	100	
15	*frequency*=0	100	How many drug related crimes have the offender involved in?
	1≤ *frequency* <4	80	
	frequency≥4	60	
21	*percentage*<10	100	The percentage of all the relations of the offender in relationship finder who have committed felony crime.
	10≤ *percentage* <60	80	
	percentage≥60	40	
22	*member*=0	60	How many family members does the offender have? The data are coming from relation finder.
	1≤ *member* <4	80	
	memeber≥4	100	
31	*Don't have a job*	40	If the offender have a job before he/she entered jail.
	Have a job	100	
41	*age* <20	20	The age when the offender be released from jail.
	20≤ *age* <40	60	
	40≤ *age* <50	90	
	age≥50	100	

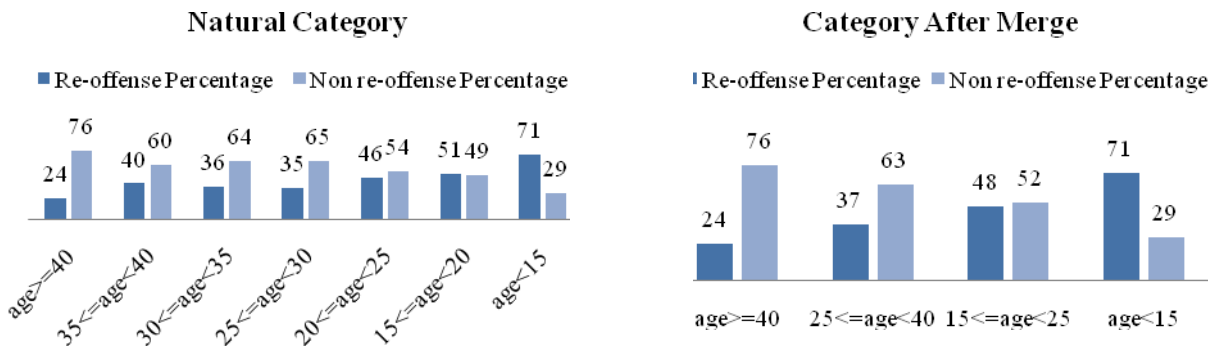


Figure 2.2. Categories of "Age of first crime" factor

2.3.3 Optimizing the factor and category weights

We make use of a genetic algorithm [17] in our optimization component so as to optimize the weight for each category and factor. It works in two steps. We first optimize the set of weights for the factors which compose an individual category. Our goal here is to achieve the best AUC score for that category based on the available historical data. The second step is to compute the optimal weights of the categories so as to achieve the best AUC score of the final score of the offender, again, based on available historical data.

Our genetic algorithm is shown in Figure 2.3. In the first step, we randomly create 50 different sets of weights for each factor and category and store them in an array. Then we compute an AUC score from the sample historical data. We define a threshold for the AUC score of 0.8 which is considered a very strong correlation of the risk level and the recidivism rate in our system. We also define a loop limitation of 10,000 rounds, and the algorithm will stop when

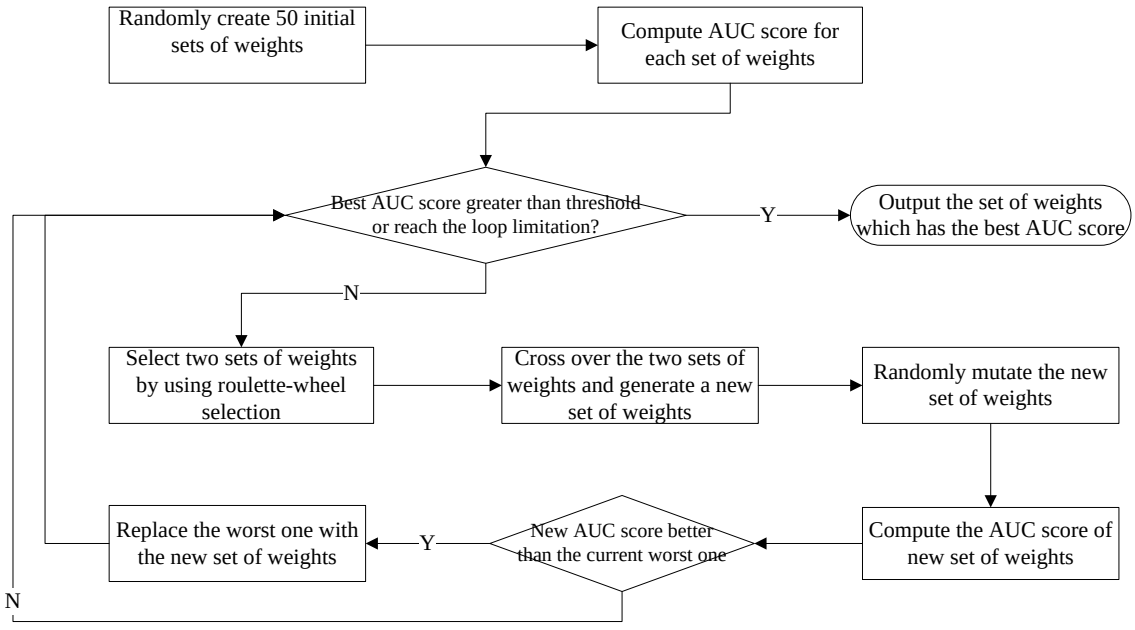


Figure 2.3. The genetic algorithm

either the threshold or the loop limitation is reached.

The core part of our genetic algorithm is creating offspring (a new set of weights) from the current inputs. In our system, we use the "roulette-wheel selection" method [4] to choose the two sets of weights to be used as parents in order to generate offspring. For each of the weights, the offspring inherit 60% gene (weight1) from the parent having the best AUC score and 40% from the other parent (weight2). So the weight of the new offspring is computed as shown in Equation 2.

$$offspringWeight = 0.6 \times weight1 + 0.4 \times weight2 \quad (2)$$

After creating the offspring, we perform a mutation on the offspring. To do this, we randomly choose two weights from the set. We increase the first chosen weight by 3 points and decrease the second weight by 3 points, thus mutation maintains our total weight.

Then we use the sample data to compute the AUC score of this new offspring. If the AUC score is better than the lowest current score in our array of 50 sets, then we replace the entry in the array corresponding to the lowest score with the new offspring. The genetic algorithm continues running until it reaches either the AUC score threshold or the loop limit. When the genetic algorithm is finished, we use the final weight of each category and factor as the weights of the system.

2.4 Experimental Results

2.4.1 Experiment Design

Our definition of recidivism in our system is that an inmate is a convicted of a felony charge within four years of being released from jail. In general, researchers tend to use three years as the period to observe re-offending, but we add one more year to allow the court sentencing data to enter our system so that we see the charge as convicted.

Table 2.3. AUC score function

ComputeAUC ()

1. *sortScoreArray* ← sort original score in decreasing order
2. *FP, TP, FPprevious, TPprevious* ← *0*
3. *AUC* ← *0.0*
4. *scorePrevious* ← -1
5. **for** *i*=0 to *sampleSize*
6. **do if** *sortScpreArray*[*i*] ≠ *scorePrevious*
7. *AUC* ← *AUC+(FP-FPprevious)×(TP+TPprev)÷2*
8. *scorePrevious* ← *sortScpreArray*[*i*]
9. *FPprevious* ← *FP*
10. *TPprevious* ← *TP*
11. **end if**
12. **if** *sortScpreArray*[*i*] has re-offense
13. *FP* ← *FP* +1
14. **else**
15. *TP* ← *TP* +1
16. **end if**
17. *AUC* ← *AUC+(N-FPprevious)×(N+TPprev)/2*
18. *AUC* ← *AUC ÷ (N × P)*
13. **Return** AUC

Here, P is the total number of positive events and N is the total number of the negative events in the sample data

The sample data that we used to evaluate the system was chosen from the inmates in Madison County Jail who are Alabama residents and were released from jail between January 1, 2004 and July 1, 2004. The total number of such inmates is 625.

The measurement used in our system to evaluate the strength of the correlation between the risk level of inmates and the recidivism rate is the AUC score. We treat recidivism as a negative event and non-recidivism as a positive event. A function to compute an AUC score is shown below in Table 2.3.

2.4.2 Experimental Results

The AUC score of the sample data is 0.73 which means that the score of an inmate has a strong correlation to the recidivism rate. We define a low risk inmate as one with a score greater

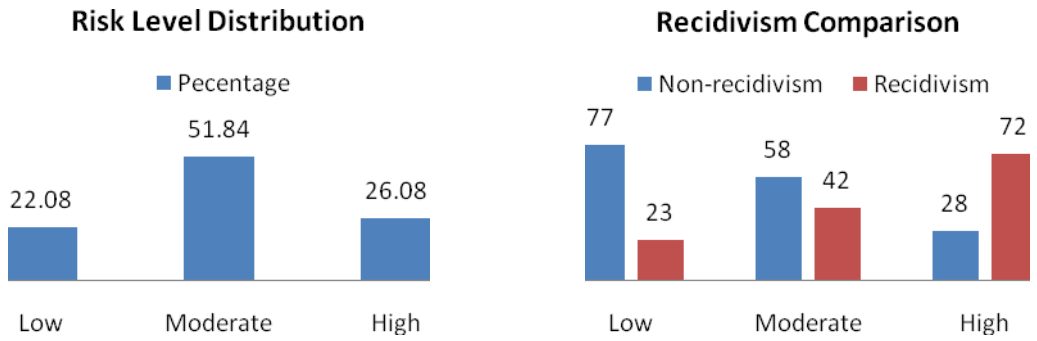


Figure 2.4. Score distribution and recidivism comparison

than 80 in our system. A moderate risk inmate will have a score between 55 and 80. When the score is lower than 55 the inmate is considered to be a high risk level.

We also analyze the risk level distribution and the compare the non-recidivism rate and recidivism rate among different risk levels. This comparison is shown in Figure 2.4. From Figure 2.4 we can see that the largest category of inmates is the moderate risk level. The numbers of high risk level and low risk level inmates are relatively similar. Looking at the recidivism comparison chart we can see that the inmates in the low risk category have a significantly lower recidivism rate. Inmates in the higher risk categories have a higher likelihood to commit crimes in the future and the risk level and the recidivism rate are roughly in linear relation.

2.5 Conclusions and Future Work

In this paper, we constructed a system named IRAS (Inmates' Risk Assessment System) to evaluate the risk level of inmates in the Madison County Jail in the state of Alabama. The risk level of inmates has been categorized into low, moderate and high. Our experimental results, which are based on 625 inmates who were released from Madison County Jail, show that the risk level suggested by our system has a strong correspondence to recidivism rate. We believe this system will help criminal justice agencies, law enforcement and corrections departments to

classify offenders and make improved decisions regarding inmates chosen for pre-trial and other release programs.

In the future, we wish to extend our current optimization component to include multiple weighting functions. This would allow us to classify inmates based on the types of crimes that they have committed and apply a different weighting function for a different type of criminal. For example, an inmate involved primarily in robbery crimes may have a different re-offending pattern from an inmate involved in drug related crimes or sexual assault. Another enhancement that we wish to consider is to define a metric for the "impact of re-offending" and use this as a factor when evaluating the risk level of an inmate. We will also continue to cooperate with the Madison County Jail to evaluate the real world performance of our system.

References

[1] D. Andrews and J. Bonta, "The Level of Service Inventory-Revised", Toronto, Ontario, Canada: Multi-Health, 1995.

[2] D. Andrews, "Recidivism is predictable and can be influenced: using risk assessments to reduce recidivism", Forum on Corrections Research, pp. 11–18, 1989.

[3] J. Austin, D. Coleman, J. Peyton, K. Johnson, "Reliability and Validity Study of the LSI-R Risk Assessment Instrument", 2003.

[4] J. E. Baker, "Reducing bias and inefficiency in the selection algorithm", Proceedings of the Second International Conference on Genetic Algorithms and their Application, Hillsdale, New Jersey, USA, pp. 14-21, 1987.

[5] R. Barnoski and S. Aos, "Washington's Offender Accountability Act: An Analysis of the Department of Corrections' Risk Assessment", 2003.

[6] R. Borum, P. Bartel, A. Forth, "Manual for the Structured Assessment for Violence Risk in Youth (SAVRY) (Consultation Ed.)", Tampa: University of South Florida, 2001.

[7] L. Ding and B. Dixon, "Using an Edge-dual Graph and k-connectivity to Identify Strong Connections in Social Networks", ACMSE 2008, Mar, 2008.

[8] T. Fawcett, "An introduction to ROC analysis", Pattern Recognition Letters, Vol 27, pp. 861-874, 2006.

[9] G. Gene and H. Kenneth, "Statistical Methods in Education and Psychology, 3rd edition", Allyn & Bacon, 1995.

[10] P. Gendreau, T. Little and C. Goggin, "A meta-analysis of the predictors of adult offender recidivism: What works!", Criminology, Vol 34, pp. 575–607, 1996.

[11] R. Hare, "The Psychopathy Checklist-Revised, 2nd Edition", Toronto: Multi-Health Systems, 2003.

[12] B. Kirkpatrick, "Exploratory research of female risk prediction and the LSI-R", Corrections Compendium: The National Journal for Corrections, Vol 24, pp. 1−17, 1999.

[13] C. Lowenkamp, A. Holsinger, E. Latessa, "Risk/Need Assessment, Offender Classifcation, and The Role of Childhood Abuse", Criminal Justice and Behavior, Volume 28:5, pp. 543-563, 2001.

[14] C. Lowenkamp, A. Holsinger, L. Brusman-Lovins, E. Latessa, "Assessing the Inter-Rater Agreement of the Level of Service Inventory Revised", Federal Probation, Vol 68:3, pp. 34-38, 2004.

[15] B. Ostrom, M. Kleiman, F. Chessman, R. Hansen, N. Kauder, "Offender Risk Assessment in Virginia: A Three-Stage Evaluation", 2002.

[16] Q. Vernon, H. Grant, R. Marnie and C. Catherine, "Violent offenders: appraising and managing risk", American Psychological Association, 1998.

[17] D. Whitley, "A genetic algorithm tutorial", Statistics and Computing, Vol. 4:2, pp. 65-85, 1994.

[18] U.S. Department of Justice website, http://www.ojp.usdoj.gov/bjs/correct.htm#findings.

CHAPTER 3

A RELATION CONTEXT ORIENTED APPROACH TO IDENTIFY STRONG TIES IN SOCIAL NETWORKS

3.1 Introduction

Social network analysis (SNA) has recently drawn considerable research attention. Scholars use SNA in various networks: virtual community networks [1]; author-coauthor networks [2, 3]; university email networks [4]; crime networks [5], etc. In the criminal justice domain, law enforcement agencies are using SNA to help investigators discover, understand and analyze the relations between criminal entities.

An important factor in SNA is the choice of how to discover interpersonal ties in social networks. Previous research on SNA can be categorized by two approaches:

- Using explicit information such as author-coauthor [2] or friends in a virtual community [1]; or
- Using implicit information to heuristically discover the ties in huge data sets [5-9].

The approach used in our system design and evaluation is based on the heuristic approach. In the example for this paper we will treat the following connections between individuals as relations:

- Sharing a common address as discovered from sources such as the drivers' license, vehicle registration, vehicle title, and citations data sources;
- Sharing a cell in jail or prison;
- Sharing a vehicle; or
- Association with the same criminal record or incident.

We will describe the integration model in detail in The Social Network Construction section.

Another important factor in SNA is the strength of the relationships -- a metric of how strong (or close) the relationships are between any two nodes in a social network. Granovetter [10] introduced the importance of weak ties, indicating that often extremely valuable information is dispersed through persons who are not close to each other. However in real-world social networks, especially in criminal networks, people tend to use more "reliable" relations to transmit sensitive information. It seems reasonable that someone who transports drugs would share delivery time information only with those who are "trusted" relations. The most recent study [11] gave two examples where removing weak ties did not destroy the information passing mechanism. Since strong ties play such a crucial role in structuring the social network, identification of the strong ties should be given special consideration.

However, in the two studies mentioned earlier, the definition of a strong tie is truly based on the explicit information given out by the people in the social network. It's hard to apply this approach to a heuristic social network. It's also subjective. People may not give the true information when they were participated in the studies, or it's hard to tell if a relation is either strong or weak by only having two choices. In heuristically constructed social networks, researchers assign a weight to each edge to indicate the strength of the relation. In a study of a criminal network [12], researchers suggest using frequency of co-occurrence of two entities as the strength of two directly connected people. If two people occurred in four different police incident reports, the highest weight will be assigned. It's a good approach to reduce the effects of noisy data such as typos or accidentally incorrect relations. However, the relation contexts between the two people have been overlooked. The four police reports may all indicate that

"Joe" and "Bob" are co-offenders in the same exact crime instance. Thus, the frequency of co-occurrence will not help to strengthen the relation between "Joe" and "Bob". Previous research efforts in this area also proposed using shortest path methods to compute the closeness of two indirectly connected nodes. However, if the original data used to build the social network graph does not have high enough quality (i.e. there are a number of incorrect connections in the network or the weights of relations are inaccurate), then the shortest path algorithm will not work well. To deal with uncertain relations and the noisy data generated by the relation discovery rules or poor quality of the raw data, one goal when measuring the strength of ties must be for the metrics to be robust towards noise.

To overcome this, we combine the graph theoretic concepts of *k*-connectivity [13] and the edge-dual graph [14], and we propose a novel measure of strength between the ties of any two nodes in the network [15]. Our measure is a quantitative and robust metric that can account for the noisy data. To do this, we translate the original social network to a corresponding edge-dual graph by extracting the relation contexts between connected node pairs and creating a new node for each unique relation context. For example, "Joe" and "Bob" may have two relation contexts by sharing two different addresses in our discovered social network. But, if they share a single address multiple times in the network, only one relation context node will be created. These new relation nodes are called context nodes. Connections are added between the new relation context nodes and their original nodes.

Figure 3.1 shows an example of a graph which presents a social network of two indirectly connected nodes A and E. Figure 3.2 shows the graph after the transformation. We call Figure 3.2 an edge-dual graph of Figure 3.1. The nodes inside the ellipse represent the relation context nodes, and the circled nodes represent persons. In Figure 3.2, node A and C share two different

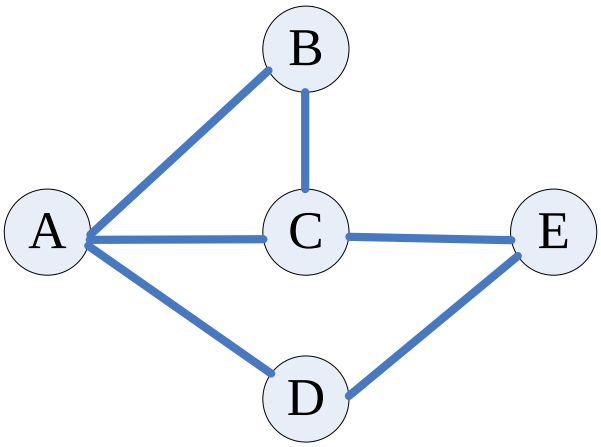


Figure 3.1. The original graph

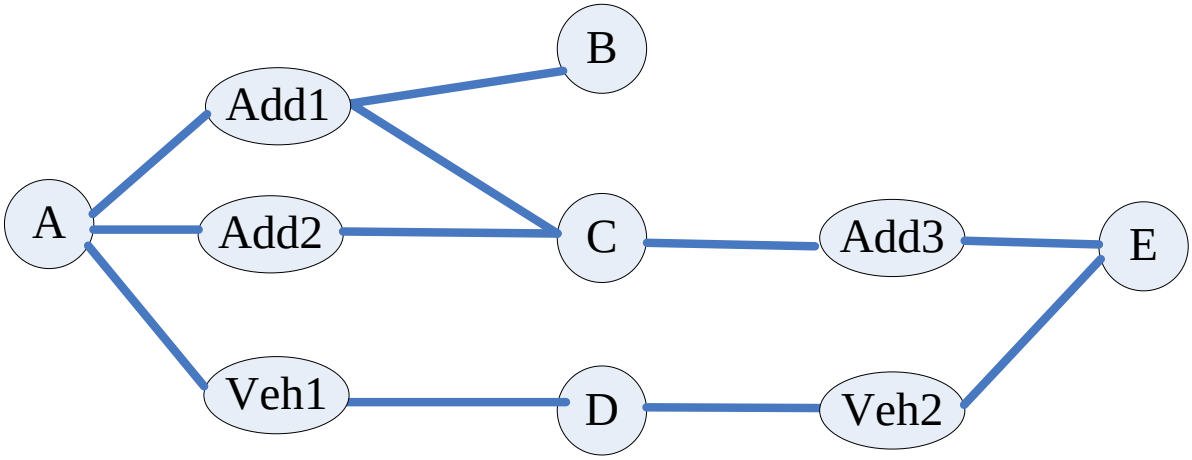


Figure 3.2. The graph after transformation.

addresses, there are two relation contexts nodes between them. Other nodes in Figure 3.2 only have one relation context node between them. After drawing the edge-dual graph we use the *k*-connectivity concept from graph theory to compute the connectivity between two nodes. Our *k*-connectivity metric measures how many relation context nodes must be removed to disconnect two given nodes. In Figure 3.2, at least two relation context nodes must be removed to disconnect node A and E. So the *k*-connectivity between A and E is 2. The most important advantage of doing this is the improvement in the SNA robustness in the presence of noisy data. In particular, a few incorrect relations will not ruin the computation of closeness. Another

advantage is that relation context nodes will help users understand the relations between nodes through visualization. Our experimental results suggest that when $k \geq 2$ the ties between the two nodes is strong.

This paper continues by presenting a literature review of related work in the next section, and then it introduces our social network construction in Social Network Construction section and the detail of our measurement in The Measurement section. The System Evaluation section describes the system implementation and experimental results. We then summarize our work and discuss the future work in a final concluding section.

3.2 Literature Review

In this section, we first review the most relevant social network construction techniques and measurements in social network analysis. We then introduce some graph theory concepts used by our algorithm.

3.2.1 Network discovery

Many different social networks have been studied by researchers. One such paper [4] analyzed a dynamic social network in a university which has 43,553 students, faculty, and staff. The ties between individuals or groups were defined as a result of examining common classes taken and email exchanges (sender and receiver). The experiment used a time window of 210 days to record the changes in the network. Their research suggests that "understanding the information and related processes in social networks requires longitudinal data on both social interactions and shared affiliations".

Other researchers [6] studied a social network based on financial transactional data records. They defined the ties between individuals by whether the individuals have a shared bank account, shared address or were involved in the same transactions. The U.S. Department of the Treasury has adapted this research to help detect illegal monetary transactions [16].

Considering criminal activity, the COPLINK system [5, 9] is based on both structured and unstructured data. The underlying social network is called the "concept space," which contains the nodes generated from source information and weighted associations between those nodes. Nodes are persons, vehicles, locations and organizations that are extracted from almost 1.5 million crime records spanning from 1970 to present. After generating the entities, they apply a technique called "co-occurrence analysis" to the concept space. This extracts relations between the entities when they appear together in the same document. The weight between the two entities was defined by the frequency with which the two entities occurred in the same report. The higher a co-occurrence weight, the more likely it is that the two entities have a strong relationship. Of the systems reviewed, COPLINK is the most similar to the approach that we are proposing.

The networks in the systems above were constructed using data from single domains (university email, financial transactions, and crime records). Our system uses different data sources coming from the state of Alabama. It includes 2003-2007 driver license registration data, 2005-2007 vehicle registration data, 1970-2007 citation data, 1958-2007 arrest data, 1994-2007 warrant data, and 1993-2007 domestic violence data. With regard to criminal data, our system contains data on all kinds and degrees of crime, including both felony and misdemeanor criminal activity. These multiple domains allow a better social network to be constructed that models the real-world social networks much more effectively.

Other researchers also studied the significant facilitators (i.e., the important variables) of discovering a social network. A study based on a covert social network involving illegal activities [7] suggested that the link formation process is influenced by a set of facilitators [8]. These facilitators may be age, race and gender [17-19] or shared affiliations between individuals

[4, 20]. The most recent research [21] compared all the factors in a crime network using Cox regression and concluded that the most significant facilitators are mutual acquaintance and vehicle affiliations. In our system, we adopt this concept and make use of all of the available addresses associated with one person (driver license registration, vehicle registration, arrest location, crime location, etc.) to create relations based on people who are connected to the same address or same vehicle.

3.2.2 Measures of Strong Ties

A social network can be considered as a graph which contains a set of nodes (individuals) and edges between the nodes which indicate the relations between the nodes. SNA researchers give several techniques based on graph theory for measuring the closeness of two nodes: frequency, shortest path search, and similarity approach.

Intuitively, if two persons contact each other frequently, they are more likely to be strong ties. Thus researchers [12] use the frequency of co-occurrence in the same police incident reports as the strength of the relation of two criminals. If the frequency of co-occurrence is greater than four, then the highest strength (weight) of the relation will be assigned. Otherwise, the strength will be reduced. As we discussed in the Introduction section, this approach fail to consider the context of relations.

Shortest path methods help users find the paths that connect two indirectly connected nodes in the social network. If this network is an un-weighted network, in which every edge is considered equally important, such as an online friend network or an email communication network, then these approaches are similar to breadth-first search [22-24]. The result can be multiple paths leading from the source node to the target node [25].

If the network is a weighted network, then a shortest path algorithm will be used in finding the highest weighted path that connect the two nodes. The highest weight will be an accumulation of each weight along the path. One research group proposed using the shortest path measurement to evaluate how close two or more indirectly connected nodes are by using a modified priority first search (PFS) algorithm [26]. The algorithm works on a weighted connected graph based on two-tree version of Dijkstra's shortest path algorithm [27], where the weight of each edge indicates the strength of the two connected entities (people, vehicles etc) within the scale of 0 to 1. Higher weights indicate stronger relations. The shortest path is the highest summarized weight by multiplying the weights of the edges discovered in the path from the source to the destination. They also compared the shortest path measurement to the breadth-first search approach and they found that the shortest path algorithm can provide more accurate relations. This algorithm depends heavily on the accuracy of the weight of the edges; a small error in the weight of an edge will cause the algorithm to produce inaccurate results. We also notice that the context of the relations have been buried in the discovery, which is quite useful for investigators. A recent study [28] from this same group applied knowledge engineering to this approach to get better results in terms of efficiency and accuracy.

Similarity methods have also been found useful in measuring the strength of the strong ties [29, 30]. These ideas are based on the observation in the paper "The Strength of Weak Ties" [10]. In that paper, Granovetter found that persons connected with strong ties tend to have similar social structures. Consider the social network as a graph, to find the strong ties, their similarities of graph structure (common neighbors and links) will be taken into account. This approach was used in social networks that only contain binary connection information [30]. In those networks, connections are created based on whether the two nodes are friends. The links in

the network are equally important. In Xiang's model [29], they allow multiple attributes and interactions between two nodes to be considered when applying similarity measurements to improve the quality of the measurement.

It is important to note that none of the approaches above address how we can give users a quantitative measurement of the closeness of the relations. They also ignore the relation context in the ties. Previous approaches use relative measures to identify strong ties. However, an improved quantitative measurement is essential when there is an extremely large set of nodes and relations forming the social network. Since this is the case in the criminal justice domain, an approach like this shows promise in helping investigators to eliminate most of the unimportant relations and better understand the relations.

3.2.3 Related graph theories

Edge-dual graph theory has been applied in the analysis of topological phase transitions [42]. An edge-dual graph, *G'*, of a graph *G* is one that has as many vertices as *G* has edges. $V' = E$ and $(e, f) \in E'$ iff (e, f) share a vertex in *G* [14]. In this paper we will utilize the concept of an edge-dual graph to abstract the relation contexts in the original graph and transform it into what we call a relation context oriented graph.

A cutset of a graph is a collection of specific nodes that, if removed, would break the graph into two or more disconnected pieces. A graph is *k*-connected (i.e., has node connectivity k) and is called a k-component if it has no cutset of fewer than k nodes [13]. In graph-theory terminology, a *2*- or biconnected component is called a bicomponent and a *3*-connected component a tricomponent.

The *k*-connectivity concept has been used in SNA research as "structural cohesion" [32] to identify the groups in a social network. The k-connectivity concept has also been adopted in

other areas, such as the evaluation of fault tolerance of sensor networks [33] and the study of graphs with geometric space [34]. From previous research efforts [32, 33] we know that one of the important characteristics of k-connectivity is its ability to be resilient to noisy data. In this paper, we modified this concept to consider only the affects of the removal of "relation context nodes" (as defined above).

3.3 Social Network Construction

Our original data sources for this research are from multiple law enforcement and other government databases from the State of Alabama. We first integrate the data sources to a central database and create relations between people by the rules we described below. We then remove relations that are likely to be false, created by heuristics that are designed to be "optimistic". After that, we create a common model for representing the social network. These steps are addressed in detail in the paragraphs below.

3.3.1 Relation creation

We use social security number as the identification of a person to integrate the information of a person from multiple data sources. Around 5% of the persons in our data sets are unidentified and will not be added to our system. The relations in our system are discovered by several rules. They can be categorized to three major relation types:

Sharing an address. When two people have the same address in the integrated data source we will generate a relation between these two people. Because we have many different data sources containing address information, there are many ways to generate an address match. For example: it is possible that person A's driver license registration address is same as person B's vehicle registration address or person C's speeding ticket's notification address. Cell mates in prison or jail are also included in this relation type.

When this type of relations is discovered, we first transform the address to the uniform form of "street number" + "street name" + "street suffix" + "apartment number." The "street suffix" part uses the USPS (United States Postal Service) standard street suffix [35]. The "street name" part is transformed to the soundex of the real name. The "apartment number" part contains only the number of the apartment, lot or building etc. By doing this, we can avoid duplicated addresses with slightly different formats or words being considered as a single address, which can affect the accuracy of our edge-dual graph construction. We also store this transformed address as the relation context between two people. We allow up to one year difference between the two records that indicate the common address. For example, family members may not change their vehicle's registration address at the same time after moving. Intuitively, records with very different dates have a greater chance of producing a false relationship than records with similar dates. By setting the tolerance to one year, we can eliminate some of the false connections in the social network.

Sharing a vehicle. Here we have two possible situations. First, person A is the registered owner (title) of a vehicle which person B licenses (registers). Second, person B received a ticket while driving person A's car. In both cases, we store the vehicle identification number (VIN) as the relation context between two persons. Since every vehicle has a unique VIN, we can use this information as the identification of this type of relation.

Involved in the same crime incident or domestic violation. In domestic violations cases, the offender has a direct relation to the victim. In criminal violations cases, we generate relations between all persons with the same role in the crime. For example, the victims in the same case will have relations of low weight. Our original data source did not contain the "offender" role of crime data. In our data sets, every domestic violation and crime incident has a

state-wide unique case number. We use the case number as the identification for this type of relation.

3.3.2 Removing false relations

In the relation creation process, false relations may be created by noisy data. In our system, this can happen when addresses contain apartment numbers or other detailed but irregular descriptions. The original data source often truncates the apartment number or other address specifying information due to length limitations. This will affect the correctness of the address match since many people may appear to live at the same address when in fact they do not.

To deal with this noisy address problem, we keep a counter for each unique address and track the number of people that live at that address. According to the U.S. Census Bureau [47], the average family size in U.S. is 3.19. We round that value up to 4. There is generally some delay in the information in our databases, e.g., when people move out a house they may not change their driver's license registration immediately. Thus, we doubled the average family size to 8 as our threshold for a normal number of persons recorded to be living at a single address. If a relation is generated using a match of an address with an abnormally large count of persons (greater than 8), we will eliminate the relations from our social network.

Table 3.1. The common relation model

Social Security Number	The social security number of person A.
Related SSN	The social security number of Person B' related to person A.
Relation Context	The relation between the two persons, such as an address, a vehicle VIN, and crime case number.
Relation Type	One of the three categories from Section Social Network Construction.

After performing these steps we can create the common relation model. In this model persons are identified by their social security numbers. Table 3.1 describes the model and detail information stored in the model.

3.4 The Measurement

To measure the strength of any two given persons in the constructed network, our algorithm will first transform the original graph to an edge-dual graph (relation context oriented graph). Then we will compute the local *k*-connectivity of the edge-dual graph. The detail algorithm is described in following subsections.

3.4.1 Edge-dual graph transform

Based on the common model generated in last section we can construct an undirected graph, *G*. The graph *G* stores the "natural" representation of the model, i.e. each vertex ($v \in V$) represents a person, and each edge ($e \in E$) of the graph represents a relationship. We store the edges of *G* by using an adjacency list. We create the edge-dual graph *G'* from *G* through the following steps:

1. Let $V' = V$.
2. Starting from any node, traverse the graph using breadth first search (BFS).

3. Let $e = (A,B)$ be an edge of G discovered by BFS, and let r be the relation context of e. Create a new node, $r \in V'$, in G', unless r already exists. Label this new node in G' as a relation context node.

4. Construct two new edges in G': $e_1 = (A,r)$ and $e_2 = (r,B)$.

The algorithm takes O($|E|$ log $|E|$) time in the worst case to transform the graph. Because each edge of G could have a distinct relation context, G' can contain as many as $|E|$ relation nodes and thus up to $|V|+|E|$ total nodes. Step 3 must determine if a relation context node corresponding to r already exists in V'. In the worst case, this will take O(log $|E|$) time to perform this search of the node set, however hashing the relation contexts can make this much faster in practice.

The graph G' is the relation context oriented graph, relation contexts of each edge in the original graph G have been extracted and relation context nodes have been created in this new graph. Figure 3.1 and Figure 3.2 illustrate the transform process. Figure 3.1 shows an original graph constructed by the common model. In the original graph, we know A, B, C, and D are related, but we don't know why they are related to each other. After the transformation, new relation context nodes "add1", "add2", "add3", "veh1", and "veh2" are added to the graph. So we know that A and B are connected because they share an address "add1". Also, since B and C also share the same address, they are connected to the same relation context node "add1". After constructing the relation context oriented graph, we can use the local connectivity concept to compute the strength of the relation of two given nodes.

3.4.2 The *k*-connectivity measurement

Recall the original definition of connectivity of a graph is the minimum number of vertices whose removal disconnects the entire graph. In this paper, we are interested in what is

known as local connectivity: given two nodes x and y, we wish to determine the number of relation context nodes whose removal will disconnect x from y. (Recall that the relation context nodes are the new nodes added to G' that contain relation context information).

This local connectivity problem can be solved by modifying traditional Max Flow algorithms [48]. Our algorithm runs as given below:

1. Given edge-dual graph *G'*, transform *G'* to a flow graph named *G''*.
2. Starting from source node *s* of *G'*, traverse the graph using breadth first search (BFS).
3. Let *e = (A,B)* be an edge of *G'* discovered by BFS. If *A* is a relation context node, split *A* to two nodes *A_i* which indicate the incoming node and *A_o* for the outgoing node; else does not split *A*. Do the same for node *B*. Then add the all the nodes to *G''*. Create a new edge *e'* = (*A_o*,*B_i*) or (*A*,*B_i*) or (*A_o*,*B*) (depending on the type of nodes), and two new edges *e1'*= (*A_i*,*A_o*) and *e2'*= (*B_i*,*B_o*) or one new edge (depending on the type of the nodes). Then add the new edges to *G''*.
4. After getting the flow graph *G''*, call the function **Find-connectivity** (*G*,*s*,*d*) to compute the connectivity of two given node *s* (which indicates the source node) and *d* (which indicates the destination node). Use the *G''* as the input of the graph.

During the above process two sub-algorithms are performed, given below as Find-connectivity and Augmenting-path:

Find-connectivity (*G*,*s*,*d*)

1. **for** each edge *e = (A,B)* ∈ *G''* assign the capacity equals 1 and current flow (*cf*) equals 0.
2. Initial connectivity *k* to 0.
3. **while Augmenting-path (G,s,d) is true**
4. do *k*← *k* +1
5. **return** *k*

Augmenting-path (*G*,*s*,*d*)

1. **for** each vertex $u \in G$
2. **do** *color*[*u*]← WHITE
3. π[*u*]← NIL
4. *color* [*s*]← GRAY
5. Q← Ø
6. ENQUEUE (Q,*s*)
7. **while** Q ≠ Ø
8. **do** *u*← DEQUEUE (Q)
9. **if** *u* ≠ *d*
10. **for** each *v* ∈ Adj[u]
11. **do if** *color*[*v*] = WHITE and edge *e*(*u*,*v*).*cf* = 0
12. **then** *color*[*v*]← GRAY
13. ENQUEUE (Q,*v*)
14. **else**
15. **while** π[*u*] ≠ *s*
16. **do** edge *e*(π[*u*],*u*).*cf*=1
17. *u*← π[*u*]
18. **return true**
19. **end if**
20. *color*[*u*]← BLACK
21. **return false**

π[*u*] is used to record the parent node of *u*

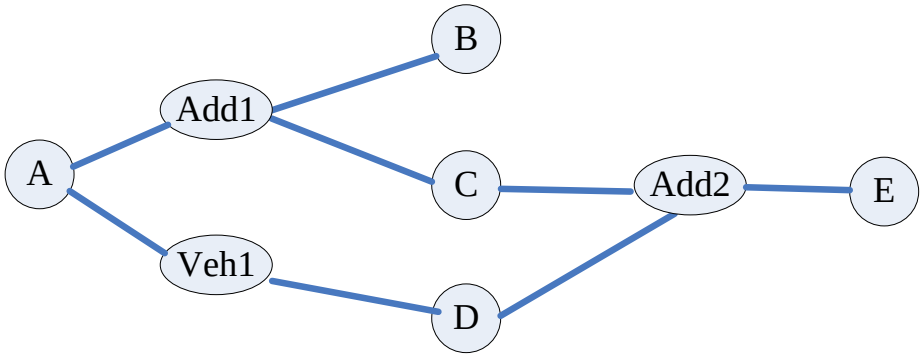


Figure 3.3. *1*-connectivity of the nodes *A* and *E*.

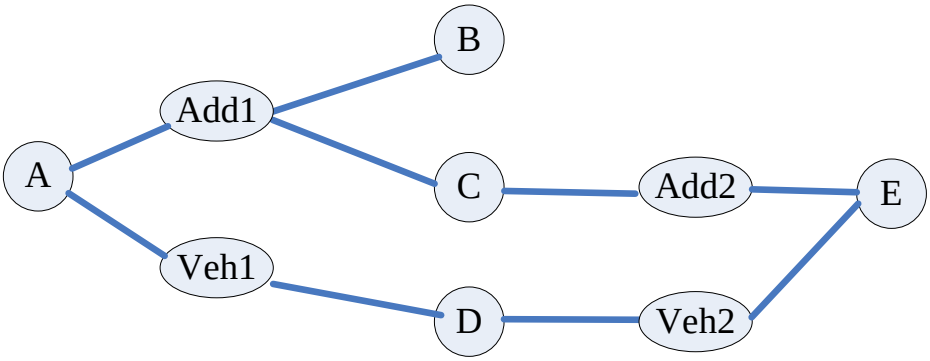


Figure 3.4. *2*-connectivity of the nodes *A* and *E*.

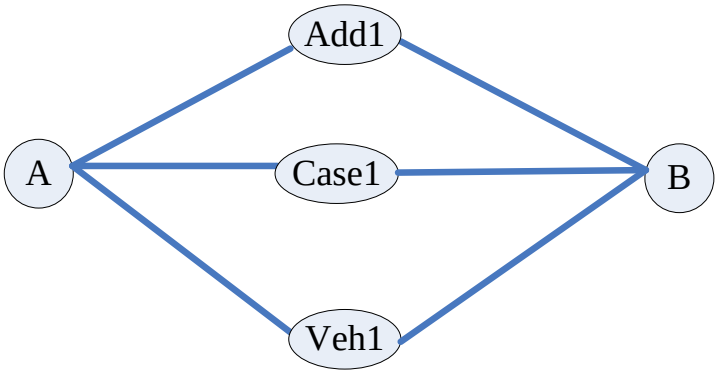


Figure 3.5. Connectivity of two directly connected nodes *A* and *B*.

Notice that transforming an edge-dual graph to a flow graph part can be done when transforming the original graph to edge-dual graph, so the cost of this operation is minimal. Therefore, the time-complexity of the algorithm depends on the running time of the function Find-connectivity, which is O(k $|E|$) where k is the connectivity of the two given vertices. In our

system, we are only trying to determine if two nodes have some specified minimum level of connectivity (such as 3). Thus for our purposes, k is a small constant, and the algorithm will run in O(|*E*|) time.

Below are two examples showing the connectivity of two indirectly connected nodes in the graph. Figure 3.3 shows a graph where the connectivity between A and E is 1. Figure 3.4 shows a modification of Figure 3.3 where the connectivity between A and E is 2. Compare Figure 3.4 to Figure 3.3, we are more confident with the relations between A and E. In Figure 3.3, the relation context node "add2" might be an error or falsely reported. The nodes A and E would no longer be connected if that is the case. In contrast, to separate A and E in Figure 3.4, we must remove at least two relation context nodes. So the relation of A and E in Figure 3.4 is stronger than in Figure 3.3.

K-connectivity can also be applied to two directly connected people. As shown in Figure 3.5, persons A and B are connected through three different relation context nodes. So their connectivity is 3 and the relation between A and B is very strong. We need to remove all three relation context nodes to disconnect A and B.

3.5 System Evaluations

As we mentioned before, the purpose of this research is to help law enforcement identify the relationships between people accurately and efficiently. Therefore we make use of as many different data sources as are available to us in order to construct the "best" real-world system. We use this system to evaluate our proposed measure of closeness of the relations between people in social networks. We describe our system in the following subsections.

3.5.1 Data Integration

As we mentioned before, our data comes from the State of Alabama. The total data size is 450GB. The ETL integration process for these databases took approximately 2 days of CPU

and disk time on a windows XP machine with 2GB memory, 2TB hard disk and 2 GHz CPU. The table of relations for the entire composite dataset has total size 40GB and 211,403,212 rows, which means that our discovered social network contains 4,906,460 nodes and 211,403,212 edges. Because this relations table is so large, we store the relationship information in an incremental fashion. We separate the relations information into two year segments, which means that every two years we will create a new table for the relations. This incremental storage method also allows us to update the data without re-computing the entire relations table.

3.5.2 Visualization Tool

To help a user intuitively understand the relations between people, we have developed a Java graphic user interface based on JGraph [38], an open source visualization tool. We also allow users to choose the maximum depth of the relations displayed.

Below are some screen shots of the visualization tool. Figure 3.6 shows the query interface of the system that allows users to input the social security numbers of two persons in interests (source SSN and destination SSN). Figure 3.7 shows the output of the original network graph of the given two persons in our system, which is typically generated in less than 20 seconds. Because our example is generated from a real world database, and because of privacy issues, we mask the real social security number of the persons involved. The node marked with "A" indicates person A or the source node and the node marked with "B" indicates person B, which is the destination node. The nodes marked with X indicate unique persons other than A or B. Figure 3.8 shows the edge-dual graph after users click the "analysis" button from Figure 3.7. In Figure 3.8, the yellow node between two persons is a context node. Every context node has a unique identifier, which is the address of a home, VIN, or a case number.

If we examine Figure 3.7, the two nodes in question (A and B) appear to be reasonably well connected. However, when we translate the graph to the edge-dual graph we find that the two persons in question can be disconnected by the removal of a single node and thus their connectivity is only 1 (surrounded by the red eclipse). In this case it means that the two persons are connected only through a single address. If this address is made by noisy data, then these two persons will not be connected in reality. This example shows the power of our measurement toward noisy data.

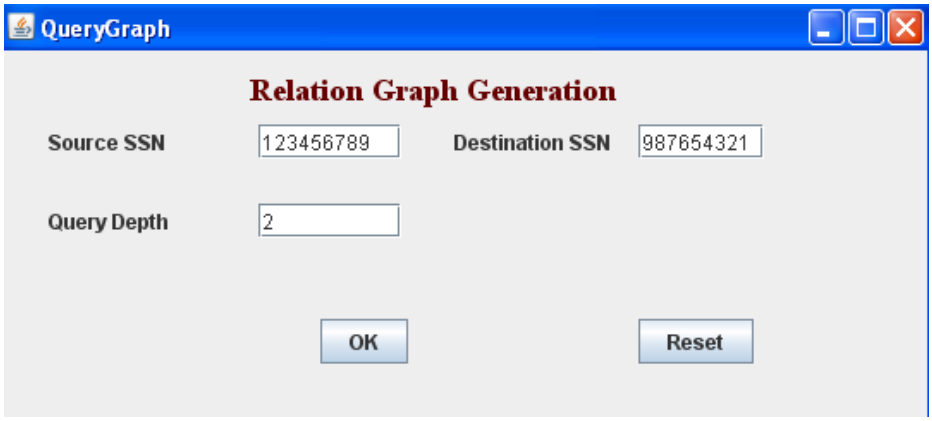


Figure 3.6. The query interface.

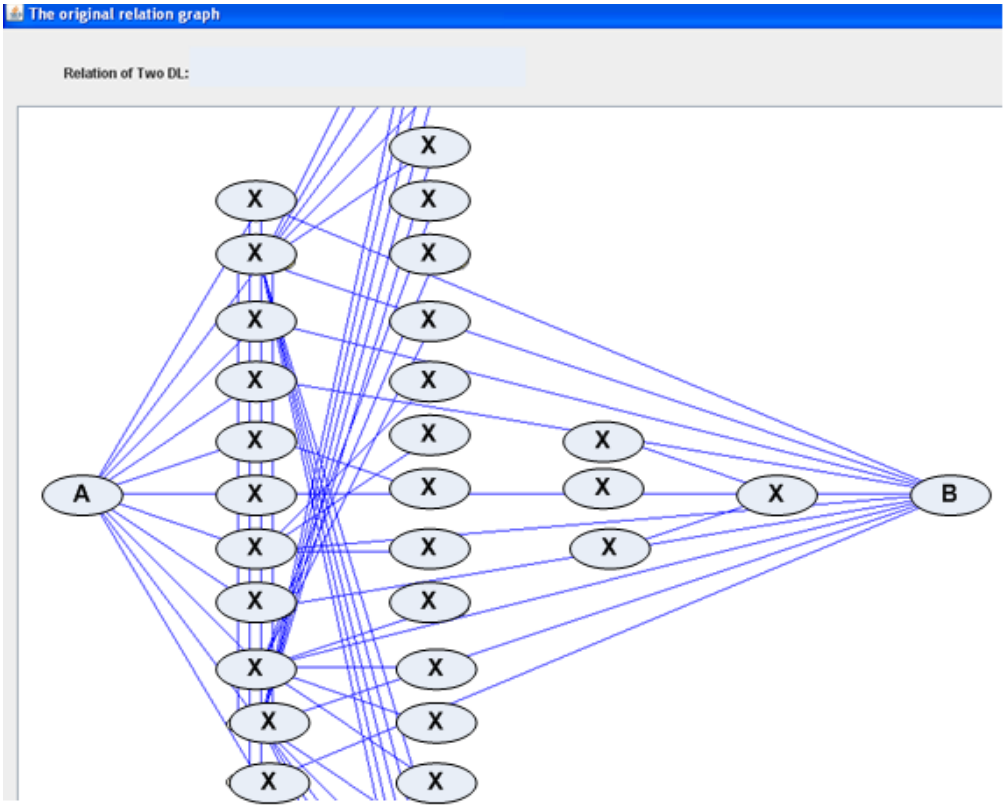


Figure 3.7. The original graph.

An additional advantage of our model for the relationships is that it first exposes relations, and then it helps the user (e.g., investigator) understand the contexts of the relations that exist between two persons. For example, an investigator learns that two persons share a car that is used in a crime from the relation graph. This information may help him/her indentify the

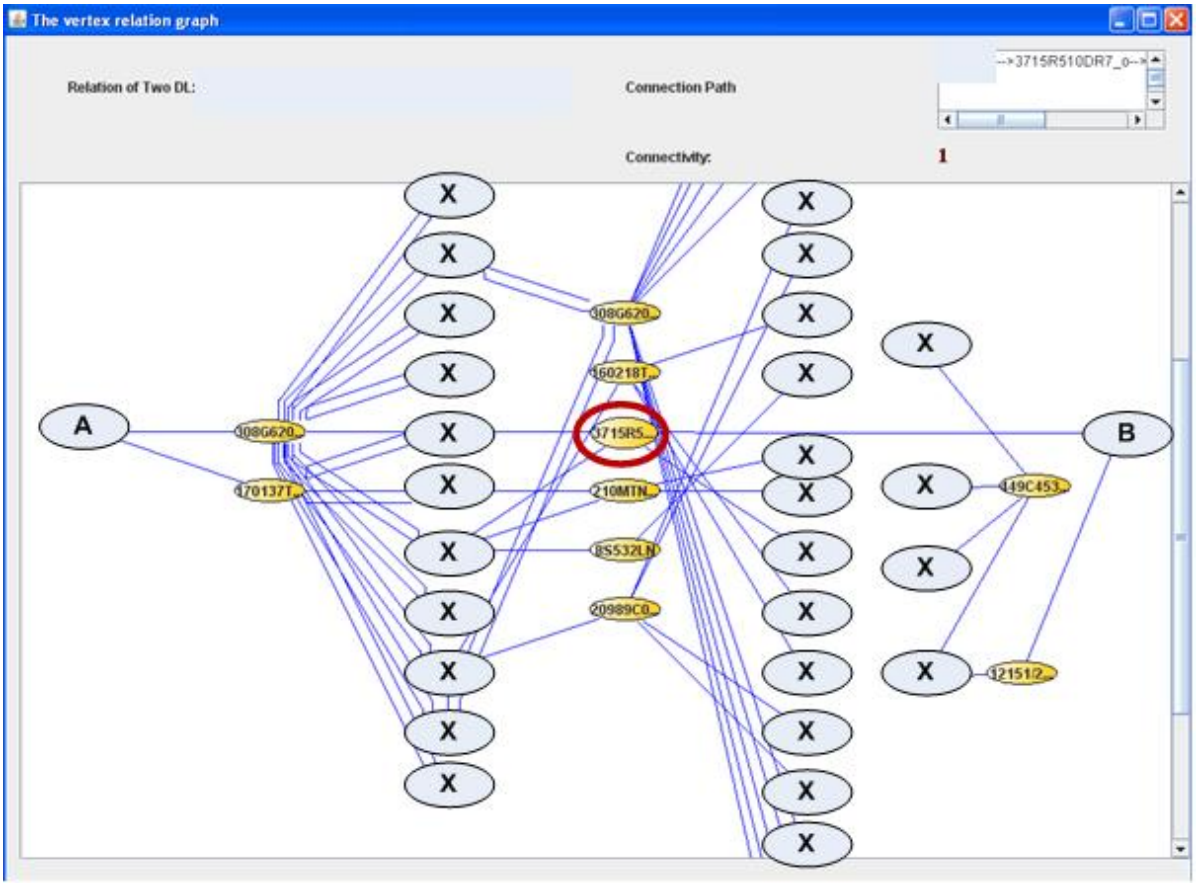


Figure 3.8. The transformed edge-dual graph.

two as co-offenders. Such applications as this would certainly help identify the co-offenders or perhaps even the gangs involved in criminal activity.

3.5.3 Experimental results and discussion

3.5.3.1 Experimental Design

To validate our approach, we treated the co-offenders who had been involved in the same robbery as strong ties in the network, and then statistically analyzed their connectivity. Previous research has shown that past robbery co-offenders tend to commit robbery together in the future [39]. We stored them in a table named co-offenders as our fundamental testing data. Because one of the purposes of this research is to help investigators find potential suspects based on historical data, we take this one step further to validate that strong ties play an important role in predicting future co-offenses. To do this, we removed the most-recent connections of these co-offenders from the discovered social network, and applied our algorithms to compute the

closeness of these offenders based on the new network (which can be considered as a historical snapshot to our discovered social network). We then checked if the high connectivity ties can predict future co-offenses better than lower connectivity ties.

Our current databases do not contain the co-offenders information directly. To infer offender relations we chose the persons who committed robbery on the same day in the same city, and with the same warrant issue date. Table 2 shows the past five year's crime statistics report of robbery in "Crime in Alabama 2007" from Alabama Criminal Justice Information Center (ACJIC) web site [40]. There were 19.2 robberies per day in 2007 state wide in this statistical report. According to the data released in this report, if we only choose the cities whose population is between 10,000 and 50,000, then there were 4.1 robberies per day in these cities, and 0.09 robberies per day per city. Based on the statistical report we can safely draw a conclusion that if two persons have committed robbery in the same city, on the same day and have the same warrant issue date, then they are very likely to be co-offenders. Our statistical analysis in the hypotheses test section also supports this assumption. A table named co-offenders is used to record all of these co-offenders involved in robbery crimes.

Table 3.2 shows that there are 2,678 pairs of persons in the past 50 years and 880 pairs of persons in the past 5 years in the co-offenders table. Most robbers are repeat offenders, which is the reason why the past 5 years co-offenders are not proportional to last 50 year co-offenders. There is an average of 176 pairs of persons per year in the co-offenders table. We also discovered all the people who have at least 1-connectivity and have committed a reported robbery in the network with the removal of those most-recent connections. We stored those relations as sample data in a table named discovered-co-offenders. We randomly chose 100 pairs of persons from the co-offenders table, 300 pairs of persons that have 2 or more

Table 3.2. Co-offenders and discovered-co-offenders tables

Total Pairs in co-offenders table (ctotal)	**2678**
Total Pairs in co-offenders table in past 5 years (c5year)	**880**
Total Pairs in discovered-co-offenders table (dtotal)	**11722**
Total Pairs in discovered-co-offenders table in the past five years (d5year)	**2541**

connectivity, and 300 pairs of persons who have 1-connectivity from discovered-co-offenders table.

The information in the co-offenders table is just a subset of all co-offenders in robbery crimes in Alabama, while the information stored in discovered-co-offenders table is all the possible co-offenders we can discover in Alabama. Based on this observation, we define a factor called Best Expectation Match Rate (BEMR) to indicate ideally how good we can predict the future co-offense in Equation 3.

$$bemr = \left(\frac{d_{total}}{c_{total}} + \frac{d_{5years}}{c_{5years}}\right) \div 2 \quad (3)$$

From the equation we get BEMR= 28.74%. Observe that even when all of the relations we discovered are correct; the best BEMR prediction match rate is only 28.74%. BEMR will be used as a benchmark in our prediction experiment.

3.5.3.2 Hypotheses and validation

H1: 2-connectivity indicates strong ties in our social network.

To test this hypothesis, we conduct two experiments. First, we randomly chose 100 known co-offenders from co-offenders table in the past 5 years and computed their connectivity. Table 3.3 shows the results of Experiment 1. We can see 66% of the co-offenders have

connectivity of "2 or more," with a fairly low standard deviation. Only 34% of them have lower connectivity.

Second, we randomly chose 300 pairs of persons who had reportedly committed robbery in the past 5 years with connectivity greater than 2 from the discovered-co-offenders table. We define coexistence in both tables as a match, so the match rate is in Equation 4:

$$match\ rate = \frac{pairs\ exist\ in\ the\ co-offenders\ table}{total\ pairs\ in\ our\ experiments} \tag{4}$$

Since our best expectation match rate is 28.74%, we define the expected accuracy as our measurement of how good these samples are in predicting the future co-offense. The expected accuracy is a relative accuracy rate compared to BEMR as shown in Equation 5:

$$expected\ accuracy = \frac{match\ rate}{bemr} \tag{5}$$

To compare the 2-connectivity ties with 1-connectivity ties, we also randomly chose 300 pairs of persons with 1-connectivity from discovered-co-offenders table to find out their accuracy. Table 3.4 shows the results of Experiment 2. We can see strong ties (*2*-connectivity) outperform weak ties (*1*-connectivity) by almost 60% (70.75% vs. 32.46%) in predicting future co-offense (p=.002).

We also run a statistical test on our assumptions of finding the co-offenders. We treat the previous mentioned 100 pairs of persons as one group (group1). We generate another 100 co-offenders as another group (group2) by creating relations between two persons who have committed same robbery in the same day in two different cities. Our test result shows that the difference of average connectivity are significant (p=.000) between the two groups. Group1's average connectivity (2.19) is significant higher than group2's average connectivity (0.74). Since our assumption of finding co-offenders creates higher connectivity pairs. If our

Table 3.3. Results of Experiment 1

Total Pairs	**100**
Average Connectivity	**2.19**
Standard Deviation	**1.26**
2-Connectivity or above	**66%**
1-Connectivity	**30%**
0-Connectivity	**4%**

Table 3.4. Results of Experiment 2

	2-connectivity or above	**1-connectivity**
Total Pairs	**300**	**300**
Average Connectivity	**2.79**	**1**
Standard Deviation	**1.06**	**0**
Match pairs	**61**	**28**
Match Rate	**20.33%**	**9.33%**
Expected Accuracy	**70.75%**	**32.46%**

assumption is wrong, then the results of matching pairs in test 2 should favor low connectivity pairs. Therefore, our assumption of finding the co-offenders stands.

H2: 2-connectivity can be differentiated from lower connectivity.

Based on hypothesis 1, we know that 2-connectivity indicates a strong tie. However it is not enough to guarantee that those relations can be differentiated from other persons who also connect to the two persons in question. For instance, person A and person B may have 2-connectivity. However, person A may have 2-connecitity with all his/her relations. In this case it becomes hard for users to choose the right relationship out of all the 2-connecity relations. We conducted another experiment to verify this hypothesis by the following steps:

1. Given two persons A and B, compute the connectivity *c* between them.

2. Compute all the connectivity between A and the persons directly related to A. Use the same method to compute all connectivity between B and his/her relations. Record the connectivity as c_i ($1 \le i \le k$) where k is the total pairs in this computation.
3. Run the experiments on the 367 pairs in Experiments 1 and 2 where the pair had connectivity of 2 or more.
4. Calculate the average below the rates in which $c_i < c$. Also, calculate the average *1*-connectivity rate where $c_i = 1$.

Table 3.5 shows the results. We can see that a large portion of relations to A or B is below the connectivity of A and B. We also run a t-test based on two groups: one group contains the connectivity of the 2-connectivity pairs themselves. Another group is the average connectivity of the persons they connect to besides themselves. Our t-test result shows that average connectivity of 367 pairs of 2-connectivity persons is significant (p=.000) higher than their remaining relations. Also otice that average 1-connectivity rate almost reaches 60%, which means that if we choose 2-connectivity as the threshold of strong ties, it will help users to eliminate many weak connections.

H3: The algorithm runs with high efficiency.

Table 3.5. Results of Experiment 3

Total pairs	**367**
Average Connectivity	**2.79**
Average Below Rate	**71.45%**
Average 1-Connectivity rate	**59.39%**
Average Persons connected to A or B	**18.92**

Table 3.6. Efficiency analysis

Database Server configuration	**2 GB Memory and SQL SERVER 2005**
Client configuration	**1 GB Memory, Windows XP Professional**
Total Pairs processed	**8252**
Total proceeding time	**16,188 (seconds)**
Average proceeding time per pair	**1.96**

As we mentioned above, our discovered social network is based on a huge amount of data, which generated 4,906,460 nodes and 211,403,212 edges. One concern about our system is the efficiency with which the relations are discovered and the connectivity between any two given persons is computed. When we performed Experiments 1, 2 and 3 we also recorded the running time of each computation. Table 3.6 shows our database server and client configuration, the total pairs being processed, and the total running time of those pairs. Our Java application is designed using up to 512M JVM memory.

The results show that the system can compute the connectivity of two given persons in an average 1.96 seconds. If we consider the whole process of displaying the original graph and the relation context oriented graph, we still get less than 25 seconds to process one pair.

3.5.4 Limitations

Our experiments have limitations such as a failure to consider other sources of data (e.g., unstructured crime incident report data, birth data, marriage data, etc.). We believe if we have more data sources the experimental results will be more accurate. Also, at this point our experiments have only analyzed co-offenses in robbery type crimes.

3.6 Conclusions and Future Work

In this paper we proposed a novel measure to identify strong ties between two persons in a real-world social network. We used an edge-dual graph transformation to abstract the relation context in a social network, and local k-connectivity between two nodes in the network to evaluate how close they are. Our experimental results show that when $k \geq 2$ the relations between two people are strong. This measure also provides a quantitative way to identify strong ties. The relation context abstracted from the relationships can provide a better understanding of the relations to the users. We believe this measure can help other researchers in the SNA area.

In the future we plan to add more relation types to our network construction, such as extracting information from unstructured crime incident reports, relations from department of corrections data, marriage license data and birth data. We plan to analyze perpetrators of other types of crime besides robbery when we gain access to additional data. We would also like to test this approach to other types of social networks.

References

[1] H. Rheingold, "The virtual community: homesteading on the electronic frontier " (The MIT Press, 2000, revised edition edn. 2000).

[2] Y. H. Said, E. J. Wegman, W. K. Sharabati, and J. T. Rigsby, "Social networks of author-coauthor relationships," Computational Statistics & Data Analysis 2008, vol. 52, (4), pp. 2177-2184.

[3] A. L. Barabasi, H. Jeong, Z. Neda, E. Ravasz, A. Schubert, and T. Vicsek, "Evolution of the social network of scientific collaborations," Physica A, 2002, vol. 311, (3-4), pp. 590-614.

[4] G. Kossinets, and D. J. Watts, "Empirical Analysis of an Evolving Social Network," Science, 2006, vol. 311, (5757), pp. 88-90.

[5] H. Chen, W. Chung, J. J. Xu, G. Wang, Y. Qin, and M. Chau, "Crime data mining: a general framework and some examples," IEEE Computer 2004, vol. 37, (4), pp. 50-56.

[6] H. G. Goldberg, and R. W. H. Wong, "Restructuring databases for knowledge discovery by consolidation and link information, Proceedings," in Proc. 1998 AAAI Fall Symposium on Artificial Intelligence and Link Analysis, Menlo Park, CA, 1998, pp.

[7] J. Raab, and H. B. Milward, "Dark Networks as Problems," Journal of Public Administration Research and Theory 2003, vol. 13, (2003), pp. 413-439.

[8] H. B. Milward, and J. Raab, "Dark Networks as Organizational Problems: Elements of a Theory," International Public Management Journal 2006, vol. 9, (3), pp. 333-360.

[9] R. V. Hauck, H. Atabakhsb, P. Ongvasith, H. Gupta, and H. Chen, "Using Coplink to analyze criminal-justice data," IEEE Computer 2002, vol. 35, (3), pp. 30-37.

[10] M. S. Granovetter, "The strength of weak ties," American Journal of Sociology, 1973, vol. 78, (6), pp. 1360-1380.

[11] X. L. Shi, L. A. Adamic, and M. Strauss, "Networks of Strong Ties," Physica A: Statistical Mechanics and its Applications, 2007, vol. 378, (1), pp. 33-47.

[12] B. Marshall, H. Chen, and S. Kaza, "Using importance flooding to identify interesting networks of criminal activity," Journal of the American Society for Information Science and Technology, 2008, vol. 59, (13), pp. 2099-2114

[13] J. Clark, "A first look at graph theory" (World Scientific, 1991. 1991).

[14] F. Harary, "Graph Theory" (Addison-Wesley, 1969. 1969).

[15] L. Ding, and B. Dixon, "Using an Edge-dual Graph and k-connectivity to Identify Strong Connections in Social Networks," in Proc. ACMSE 2008, Auburn, Alabama, US, 2008, pp.

[16] H. G. Goldberg, and R. W. H. Wong, "Restructuring transactional data for link analysis in the FinCen AI system," in Proc. 1998 AAAI Fall Symposium on Artificial Intelligence and Link Analysis, Menlo Park, CA, 1998, pp.

[17] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Homophily in Social Networks," Annual Review of Sociaology, 2001, vol. 27, (2001), pp. 415-444.

[18] A. J. Reiss, "Co-offender Influences on Criminal Careers," 1986.

[19] S. L. Feld, "Social Structural Determinants of Similarity among Associations," American Sociological Review 1982, vol. 47, (1982), pp. 797-801.

[20] L. Backstorm, D. Huttenlocher, J. Kleinberg, and X. Lan, "Group Formation in Large Social Networks: Membership, Growth and Evolution," in Proc. 12th ACM SIGKDD International Conference on Knoeledge Discovery and Data Mining, Philadelphia, PA, USA, 2006, pp. 44-54.

[21] S. Kaza, D. Hu, and H. Chen, "Dynamic Social Network Analysis of a Dark Network: Identifying Significant Facilitators," in Proc. IEEE Intelligence and Security Informatics New Brunswick, NJ, USA, 2007, pp. 40-46.

[22] M. Gordon, R. K. Lindsay, and W. Fan, "Literature-based discovery on the World Wide Web," ACM Transactions on Internet Technology, 2002, vol. 2, (4), pp. 261-275.

[23] R. K. Lindsay, and M. D. Gordon, "Literature-based discovery by lexical statistics," Journal of the American Society for Information Science, 1999, vol. 50, (7), pp. 574-587.

[24] D. R. Swanson, and N. R. Smalheiser, "An interactive system for finding complementary literatures: A stimulus to scientific discovery," Artificial Intelligence, 1997, vol. 91, (2), pp. 183-203.

[25] F. Das-Veves, E. A. Fox, and X. Yu, "Connecting topics in document collections with stepping stones and pathways," in Proc. 14th ACM International Conference on Information and Knowledge Management, Bremen, Germany, 2005, pp. 91-98.

[26] J. J. Xu, and H. Chen, "Fighting organized crimes: using shortest-path algorithms to identify associations in criminal networks," Decision Support Systems, 2004, vol. 38, (2004), pp. 473-487.

[27] E. Dijkstra, "A note on two problems in connection with graphs," Numerische Mathematik, 1959, vol. 1, (1959), pp. 269-271.

[28] J. Schroeder, J. J. Xu, H. Chen, and M. Chau, "Automated criminal link analysis based on domain knowledge," Journal of the American Society for Information Science and Technology, 2007, vol. 58, (6), pp. 842-855.

[29] R. Xiang, J. Neville, and M. Rogati, "Modeling Relationship Strength in Online Social Networks," in Proc. International World Wide Web, Raleigh, North Carolina, USA, 2010, pp.

[30] D. Liben-Nowell, and J. Kleinberg, "The Link Prediction Problem for Social Networks," Journal of the American Society for Information Science and Technology, 2007, vol. 58, (7), pp. 1019-1031.

[31] I. Derényi, I. Farkas, G. Palla, and T. Vicsek, "Topological phase transitions of random networks," Physica A, 2004, vol. 334, (3-4), pp. 583-590.

[32] J. Moody, and D. R. White, "Structural Cohesion and Embeddedness: A Hierarchical Concept of Social Groups," American Sociological Review 2003, vol. 68, (2003), pp. 103-127.

[33] E. Kranakis, D. Krizanc, and E. Williams, "Directional Versus Omnidirectional Antennas for Energy Consumption and k-Connectivity of Networks of Sensors," Principles of Distributed Systems, 2005, vol. 3544, pp. 357-368.

[34] M. A. Abam, M. de Berg, M. Farshi, and F. Gudmundsson, "Region-Fault Toerant Geometric Spanners," in Proc. eighteenth annual ACM-SIAM symposium on Discrete algorithms, New Orleans, LA, USA, 2007, pp. 1-10.

[35] http://www.usps.com/ncsc/lookups/abbr_suffix.txt

[36] http://factfinder.census.gov/servlet/SAFFFacts

[37] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, “Introduction to Algorithms” (MIT Press, 2001, Second edn. 2001).

[38] http://www.jgraph.com/

[39] A. J. Reiss, J. A. Roth, and N. R. Council, “Understanding and preventing violence” (National Academies Press, 1996, 6th edn. 1996).

[40] http://acjic.state.al.us/crime.cfm

CHAPTER 4
FIRST: FRAMEWORK TO INTEGRATE RELATIONSHIP SEARCH TOOLS

4.1 Introduction

Information technologies such as data mining [1, 2] and social network analysis [3] have been adapted to allow law enforcement to investigate crimes. Recently, geographic profiling, a technique to identify where likely suspects live, has drawn much interest from both the law enforcement and research arenas. Motivated by the advantages of these techniques in tackling crime, we have proposed a prototype crime-solving system called PerpSearch [4]. In this paper, we extend the prototype system to an integrated crime detection framework and evaluate the framework using real burglary and robbery crime reports from the city of Anniston, Alabama.

FIRST is an inter-disciplinary framework that integrates the before mentioned technologies to aid crime investigators. Given the location of a crime, with or without physical descriptions of suspects (personal characteristics and vehicle descriptions), to solve the crime, FIRST will process the inputs with following four steps as shown in Figure 4.1:

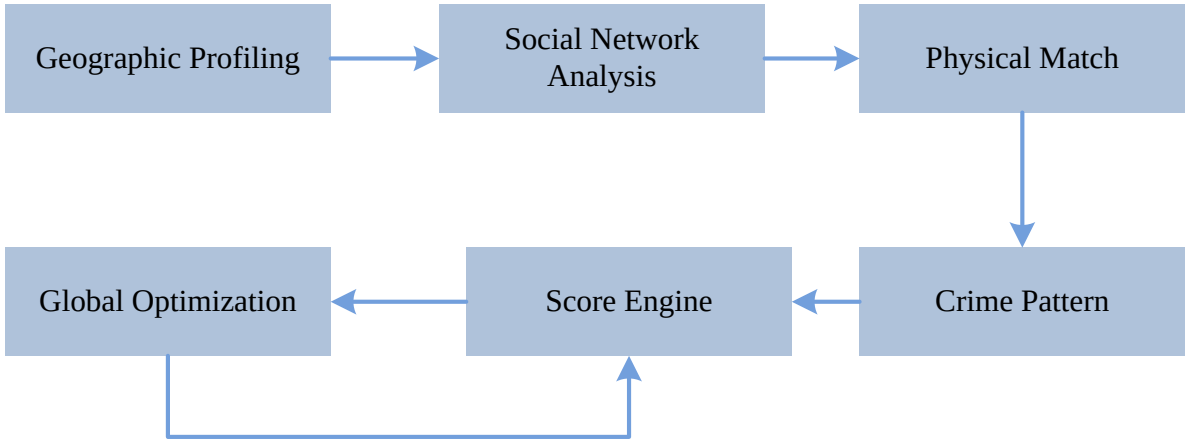


Figure 4.1. The overall searching process of FIRST

1 Apply a geospatial search based on officer selected criteria, such as radius or designated regions, or a default radius, which varies by the size of the city where the crime happens to get all qualifying addresses.

2 Retrieve all persons who have committed at least one felony crime before this crime and either are related to the selected addresses or have a close relationship to those addresses.

3 Use biometric filtering techniques to reduce the perpetrator search space. Specifically, a fuzzy search returns the persons who fit the physical description within some given tolerances. For example, given the search criteria "a 6 foot male", the search engine will return all the males within the height range 5' 7'' to 6' 3'' to avoid the potential of missing suspects.

4 Use a crime pattern component to rank the qualified suspects based on their criminal history similarities to the crime pattern of current type of crime. Statewide historical crime data has been used to analyze crime patterns of different type of crimes. A modified decision tree model is used to score suspects. For example, a person with robbery or burglary charge records is considered to have a higher probability to commit robbery than a person without those charge records.

After using the four components to process the input criteria, our score engine, which is optimized by an optimization component, gives each individual a score based on their geographical locations and crime pattern match. The officer is then given a list of suspects ordered by their scores.

To build the framework, we utilized a statewide system for law enforcement in Alabama called LETS (Law Enforcement Tactical System) [5] to get the fundamental data needed by FIRST. LETS integrates cross-jurisdictional data such as arrest records, sentencing records, drivers license registration information, vehicle registration information, as well as prison and jail information etc. With LETS data, we are able to access the geographic locations, physical descriptions, and criminal records of any individual in the state of Alabama.

We also use a social network analysis system, Relation Finder [6], to extend the ability of FIRST to find the suspects who have relations to a qualifying address near the crime or to vehicles used in the crime. Here we give a concrete real example of how social network analysis can affect crime investigation. A homicide investigation found that a white Ford truck was seen leaving a homicide scene in Crenshaw County, Alabama in October 2008. Later, after a suspect was identified, it was found that a truck matching this description was registered under the name of the suspect's father. In this example, the father and the son have a relationship in the social network produced by Relation Finder, and the suspect would be missed if we only used the current vehicle registration information to search for the perpetrator.

The primary contribution of this work is the introduction of the first (to the best of our knowledge) framework to integrate spatial analysis, social network analysis, crime pattern analysis, and biometric matching to assist in crime investigation. In addition:

- The flexibility of the framework permits replication in other states or federal agencies.
- The decision tree model of crime pattern analysis can be applied to other states with minor modification. Since most states have electronic arrest records, and

have similar crime charge codes, the model proposed here can be easily adopted into their systems and thus have a broader impact.

- Social network analysis and geographic analysis help officers expand the search beyond simply those individuals who have addresses proximal to the crime.

4.2 Literature Review

We review the most relevant research works in this section and also identify some limitations of those previous works.

4.2.1 Geographic Profiling

The concept of geographic profiling is the use of the geographic locations of crimes to give a probabilistic estimate of the location of the offender. Several researchers have used this concept to assist law enforcement investigations in different types of crimes [7-9]. There are several functions to compute the probability of offense by taking travel distance to crime scene as a parameter. Recent studies compare the performance of different functions used in prioritizing offenders in serial killings [10] and burglaries [11]. Both studies found that a logarithmic function has the best results. In this paper, we also use a logarithmic function to compute the geographic profiling score.

Geographic profiling is a useful tool to help answer the question "where," but it does not help answer the question "who?" In this paper, we apply the geographic profiling concept to FIRST, and combine it with social network analysis (SNA) and crime pattern analysis to answer both questions "where" and "who".

4.2.2 Social Network Analysis

There are numerous social network analysis studies in the crime detection area. One of the most similar approaches to FIRST is the COPLINK system [12]. COPLINK is used to detect

the relations in a criminal organization. The underlying social network in the COPLINK system is called the "concept space," which contains the nodes generated from source information and weighted associations between these nodes. Nodes are persons, vehicles, locations and organizations that are extracted from almost 1.5 million crime records spanning from 1970 to present. After generating the entities, they apply a technique called "co-occurrence analysis" to the concept space. This extracts relations between the entities when they appear together in the same document. The weight between the two entities is defined using the frequency that the two entities occur in same report. The higher a co-occurrence weight, the more likely it is that the two entities have a strong relationship.

Researchers have also studied the significant facilitators involved in discovering social networks. Several studies suggest that the link formation process is influenced by shared affiliations between individuals [13-15]. In a recent publication [16], the author compared all the factors in a crime network using Cox regression and concluded that the most significant facilitators are mutual acquaintance and vehicle affiliations. In our system, we adopt this concept and make use of all of the available addresses and vehicles associated with one person (using driver license registrations, vehicle registrations, arrest locations, and crime locations) to create relations based on people who are connected to the same address or vehicle.

All of the systems reviewed had networks constructed based on specific domains—in most cases crime data. FIRST contains not only the crime data, but other non-criminal data such as driver license registration, vehicle registration, etc. Even for criminal data, our system contains a variety of offenses, such as speeding, DUI, robbery, murder, burglary, etc. The multiple domains allow us to construct better social network models, which are closer to real-world social networks.

4.2.3 Decision Tree

Decision tree [17] is a widely used data mining technique. In general, it breaks up a complex decision into several simpler sub-decisions. In doing so, it implicitly form a tree structure that has one root node, several levels of internal node and some leaf nodes. Internal nodes contain decision rules based on one or more attributes associated with the data. Leaf nodes are decisions the tree model made. Decision tree is a good model in classification. In this paper, we modified the decision tree model not to predict outcomes or make decisions but rather to score the suspects at each leaf node.

4.2.4 Visualization Tools

Visualization is important for presenting discovered knowledge to end users. There are two separate categories of visualization tools: those that can provide the geographic view of crimes, and those that provide visualization ability for social networks. To our knowledge, only FIRST combines these two approaches.

Spatio-Temporal Visualization (STV) [18] is an example of several similar tools based on the ESRI platform [19], which can display the crime information (location, crime type and date) on a GIS map. This tool also displays some statistical results based on the date to help an investigator or patrol planner better understand the characteristics of crimes. Other tools also analyze hotspots of crime based on the frequency of crimes in specified areas.

CrimeSatIII [20] is a tool that allows the user to conduct analyses regarding spatial distribution of data, as well as analyze the spatial characteristics of data and the behavior of potential serial offenders.

Another category of tools consists of those used to visualize social networks. COPLINK [1, 18], as noted earlier, can generate a chart to display the network by traversing the graph

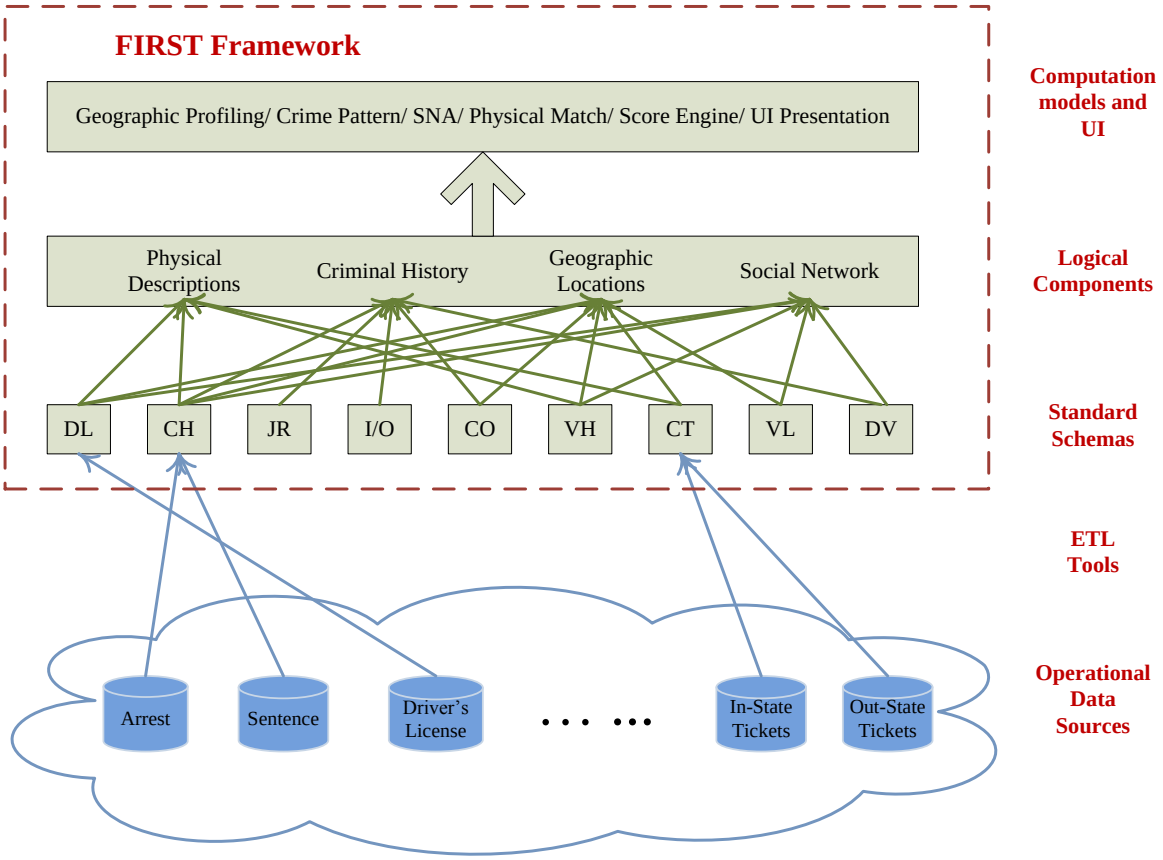
starting from a known node (person). The links between people have weights that indicate the strength of their relations. The graph is generated automatically from the underlying data source

Note that the above visualization tools lack integration. The investigator must combine the tools together to have a global view of the crime. For FIRST, since all of the addresses are geocoded, and all of the crime history information is integrated, we can display our relationship graph based on geographic information. We can also adopt the CrimeSat concept, in that we can display all the potential suspects that have ties within that area, not only by their residential address, but also based on strong relationships to proximal addresses discovered by our SNA tools.

4.3 FIRST Framework

The goal of the FIRST framework is to extract data from different data sources and to use the integrated data to help detect perpetrators. One concern of the framework is flexibility. Other states and agencies may not have the same data as ours. So we build the FIRST framework in a pay-as-you-go fashion, which means only a minimum number of components are required to construct the framework. When more data sources are available in the future, they can be easily added to the framework.

Figure 4.2 shows the current architecture of FIRST. Our data sources are coming from police arrests records, court sentencing data, juvenile sentencing data, prisons and jails data, driver license registration, vehicle registration, and traffic citations. This is our foundation layer. Data is then integrated into our logical components using ETL tools. In our system, the data sources are all stored in relational databases. The ETL tools only need to extract data from the data sources and load them to nine predefined standard schemas. As we mentioned earlier, we perform this loading process in LETS.



DL: Driver's License; CH: Criminal History; JR: Juvenile Sentence Records; I/O: Uniform Incident Report; CO: Corrections Data
VH: Vehicle Registration Records; CT: Citations; VL: Visitor Logs; DV: Domestic Violence Records

Figure 4.2. System architecture of FIRST

The nine schemas contain pre-defined fields that are needed by the logical components level. For example, the driver license schema contains unique name identification, normalized address, years the person lived in this address, height, weight, birth date, hair color, and eye color information. Information stored in this schema is supplied to the physical descriptions component, the geographical locations component and social network construction.

Of the nine schemas, only the inclusion of the criminal history schema is mandated. It is the only schema that provides information to all four logical components. Most police arrest reports contain the arrestee's identification (social security number) and physical descriptions such as weight, height, and age. It also includes at least one crime charge code against the

criminal. Charge codes used in our system are Alabama crime charge codes. Other schemas can be added to the framework later when data sources are available. The more schemas used in the system, the more reliable the returned by the system.

One consideration of decoupling the standard schema level and logical components level is the flexibility of the framework. In this framework, when an additional data source is added to the system, the only action required to do is the implementation of an ETL process to load the data source to one of the 9 schemas. The mapping between schemas and upper level components has already been defined. The data in the standard schemas will be loaded automatically into the logical components without any modification.

The four logical components provide the data needed by our computation models. In detail:

Physical Descriptions component gathers all physical description information of one person from different schemas. These include physical descriptions of a person and descriptions of vehicles that belong to the person. Physical descriptions include weight, height, hair color, eye color, date of birth, photo, tattoo, and alias. Descriptions of vehicle include year, body color, top color, bottom color, model, maker, body style, VIN, license plate number, purchase date, and registration year. One of many advantages of integrating information from different data sources is that we eliminate some potential for not returning the suspects whose physical descriptions fit a given criteria. For instance, a person's weight may changes over time. Weight of a person on his/her driver license registration may change later when he/she is arrested for some criminal charges. In our system, the two weights are both recorded in our database. If we only store one of them, when a search is performed against the database, we may miss this person.

Criminal History component contains all criminal charges of a person from his/her juvenile charges. It includes charge codes, offense date, arrest date, narrative descriptions of offense, and conviction date. In current criminal history component, we didn't include a structural description of the M. O. of a criminal. Most police agencies put the M. O. in their Uniform Crime Reports (UCR) which is a text file. It's very hard to extract information from the text file and to form structural data content. In this system, M. O. will be included in the narrative descriptions of offense field.

Geographical Locations component extracts all location information from different data sources and geo-codes them as coordinates. This component contains a street address, city, state, zip code (5 digits), years of residence, latitude, and longitude. ESRI [19] geo-coding service is used in our system to normalize and geo-code the addresses.

Social Network component stores the possible relationships discovered by Relation Finder. As we mentioned in our previous paper [6], a relationship contains the identification of a person and the identification of the person related to him/her, the type of the relations (share an address, or vehicle, or co-offenders), relation context (an address, a vehicle, or a case number). The relation context field will link to the physical descriptions component to extend the ability of FIRST to not only search for people who fit the physical descriptions, but also to search for the people whose relationships fit the input criteria.

After data is loaded to the logical components level, our computation model can take over to solve a given crime by going through the process described in Section 1.

4.4 Crime Pattern Discovery

Most research in crime pattern detection area is focused on M. O. analysis. In practice, it's hard to get all crime reports in a certain area to analyze M. O. As we mentioned before, UCR reports are often written in paper format, most police agencies don't have electronic

version of UCR reports. Among those who have electronic reports, more often they are in text file format and it is hard to do a precise analysis using this type of data source. In this paper, we put our efforts into analyzing patterns in historical crime charges against offenders. In particular, we are interested in how previous charges affect the probability of certain types of future offenses. To do so, we propose a modified decision tree model to assign a possibility score to an offender based on his/her criminal history.

4.4.1 Decision Tree Construction

In this section, we use a burglary crime as an illustration of constructing a decision tree. The construction of decision trees for other type of crimes is similar to this one. Figure 4.3 shows the decision tree we construct. We show the details to build this decision tree in following steps:

Sample Selection. The first step in constructing a decision tree is to select sample data. For a given type of crime, e.g. burglary, we choose all offenders in the state of Alabama whose last burglary offenses are within six years and who also have a felony charge before the last burglary offenses. The study of repeat offenders [21, 22] suggests that most criminals are repeat offenders who have at least one prior felony charge.

After applying the two rules to our data set, we have 8,942 qualified burglars. Of those, 85% are male and 15% are female. In addition, 8% of them are under 18, 41% of them are between 18 and 25, and 51% of them are older than 25.

The First Decision. The root node in a decision tree is the first splitting of the sample data based on a categorized attribute. Year from last felony offense date is an important attribute to predict re-offense. In the research conducted by Bureau of Justice Statistics in 2002 [23] suggest that 66% percent of prisoners who were released from prison in 1994 commit another crime within three years. So our criterion for the root node is whether an offender has committed a felony crime in the three years immediately preceeding the last burglary offense. In the sample data, 78% of the burglars have committed a felony crime before the last burglary offense and 22% of them have not. For those who fall into the 22% portion, we create a leaf node for them.

The Second Decision. After applying the first decision to the sample data, we determine the most frequently occurred charges of the remaining offenders (prior to the last burglary offenses). As shown in Equation 1, we calculate the frequency ($f_{burglar}$) of a charge code by using

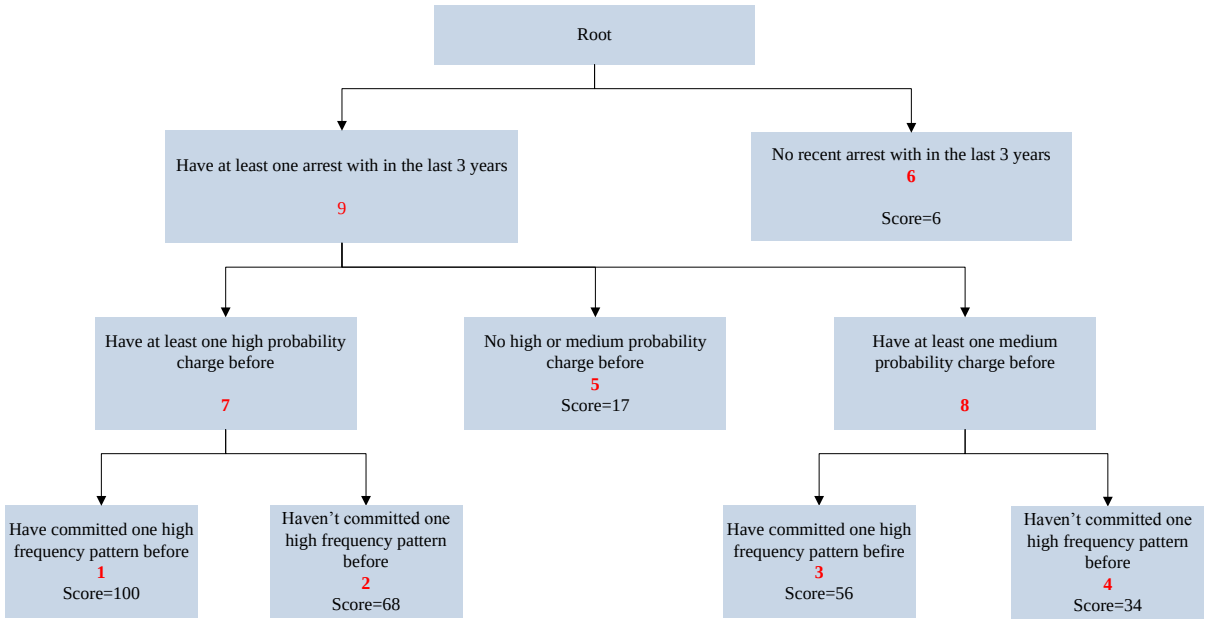


Figure. 4.3. Decision tree model of crime pattern discovery

the number of offenders who have committed a certain charge divided by the total number of burglars (6,974).

$$f_{burglar} = 100 \times \frac{\text{number of burglars with a certain charge before last offense}}{\text{total number of buglars}} \quad (1)$$

$f_{burglar}$ is only the portion of burglars who have a certain charge. It's possible that many other offenders (robbers, thieves, and murders) also have the same charge code in their criminal careers. Thus, the charge code can't be treated as a characteristic associated only with burglars. To be able to precisely compute the impact of a charge code, we introduce another measurement of the frequency called f_{all} which is calculated by Equation 2.

$$f_{all} = 100 \times \frac{\text{number of burglars with a certain charge}}{\text{total number of offenders with a certain charge}} \quad (2)$$

To compute f_{all}, we us e the same technique used in selecting sample data to select offenders with a certain charge. They must satisfy the two criteria: 1) the last known offense must be a felony charge and within six years; 2) they must also have at least one felony charge before their last felony offenses.

We then use harmonic mean to balance the two measurements. The mean f_{mean} is calculated as Equation 3.

$$f_{mean} = \frac{2 \times f_{burglar} \times f_{all}}{f_{burglar} + f_{all}} \quad (3)$$

Table 4.1 shows the portion of the results of the three measurements of burglary. We only show the charge codes whose f_{mean} are greater than 10 percent. We categorize the related charge codes to high probability, medium probability, and low probability (< 15%) charges by using an unsupervised version of the discrete filter in Weka [24]. Based on the three categories,

Table 4.2. Most frequent charge codes related to burglary, ordered by f_{mean}, given as a percentage.

Charge Codes	$f_{burglar}$	f_{all}	f_{mean}	Category
BUR3	44.28	49.88	46.91	High
TOP3	20.96	34.33	26.03	Medium
TOP2	22.97	28.33	25.37	Medium
TOP1	19.68	27.01	22.77	Medium
HARA	12.77	25.89	17.1	Low
QVOP	11.49	32.73	17.01	Low
BEMV	11.45	30.58	16.66	Low
ASS3	11.56	24.79	15.77	Low
DISO	10.93	26.10	15.4	Low
VAPM	13.87	15.71	14.73	Low
VDR1	10.00	12.73	11.2	Low
VPCO	10.60	9.64	10.1	Low

the second decision is whether the offender has at least one high probability charge or medium probability charge. If an offender falls into one of the two categories, we will further split the tree based on the third decision. Otherwise we create a leaf node for those who only have low probability charges.

The Third Decision. The third decision is based on the further analysis of high and medium probability charges. For example, given that an offender has a high probability charge BUR3, what are other high frequent charges associated with him/her besides BUR3? To answer this question, we choose the charge code (e.g. BUR3) and combine it with any charge code whose $f_{burglar}$ score is greater than 5%. Lower scores will not have enough impact on the patterns. We call this combination (charge code 1, charge code 2) a pattern. We then categorize these patterns to high frequency patterns and low frequency patterns to support the third decision.

Table 4.2. Patterns related to burglary, order by f_{mean}, in percentage.

Charge Code 1	Charge Code 2	$f_{burglar}$	f_{all}	f_{mean}	Category
BUR2	BUR3	77.6	63.18	69.65	High
CRM2	BUR3	77.6	59.48	67.35	High
BUR1	BUR3	77.6	56.64	65.49	High
JUNG	BUR3	77.6	55.58	64.77	High
CRM3	BUR3	77.6	55.46	64.69	High
PINT	BUR3	58.4	42.70	49.33	Low
VAPM	BUR3	58.4	39.53	47.15	Low
ASS2	BUR3	58.4	39.36	47.02	Low
PREV	BUR3	58.4	30.46	40.03	Low

Similar to the measurement used in the second decision, we use $f_{burglar}$, f_{all}, and f_{mean} to measure the frequency of a pattern. As shown in equation 4, fall is the percentage of offenders who have a certain pattern that commit a burglary crime in the future. The criterion for selecting offenders is the same as the method mentioned in the second decision.

$$f_{all} = 100 \times \frac{\text{number of burglars with a certain pattern}}{\text{total number of offenders with a certain pattern}} \quad (4)$$

Unlike the $f_{burglar}$ measurement used in the second decision, which is based on one single charge code, $f_{burglar}$ in this decision is based on a group of patterns. We break the sample data into two groups based on fall by using the same technique in Weka. Our $f_{burglar}$ measure is illustrated in Equation 5.

$$f_{burglar} = 100 \times \frac{\text{number of burglars with a group of patterns}}{\text{total number of burglars}} \quad (5)$$

The f_{mean} measure we use in this decision is the same as the second decision. Table 4.2 shows the patterns associated with a high probability charge code BUR3 and their frequency scores. We then further categorize the patterns to two categories based on their f_{mean} values.

4.4.2 Scores of Leaf Nodes

Figure 4.3 shows the decision tree we constructed for burglary crime after the four steps. In this decision tree, we label each leaf node with a unique number from one to six. The purpose of the decision tree is helping FIRST to better rank the suspects based on a probability score. We give the initial score of each leaf node by following rules:

1. $score_1$=100
2. 1< $score_1$: $score_2 \leq$ (median of f_{mean1}):(median of f_{mean2})
3. 1< $score_2$: $score_3 \leq$ (median of f_{mean7}):(median of f_{mean8})
4. 1< $score_3$: $score_4 \leq$ (median of f_{mean3}):(median of f_{mean4})
5. 1< $score_4$: $score_5 \leq$ (median of f_{mean8}):(median of f_{mean5})
6. 1< $score_5$: $score_6 \leq$ (Percentage of *node 9*):(Percentage of *node 6*)
7. $score_6 \geq 0$

Where $score_i (1 \leq i \leq 6)$ represents the score of a node i in our decision tree and $f_{meanj} (1 \leq j \leq 9)$ represents the f_{mean} score computed in our second and third decision level.

The rule starts with assigning the highest score 100 (FIRST uses a 100 scale score system) to node1 which represents the offenders who have committed a high probability charge within three years and have committed a high probability pattern of crimes within three years. The ratio of the score of node1 and the score of node2 represents the ratio of the probability we can predict the offender being a burglar with high probability pattern and without high probability pattern. The probabilities are the mean value of f_{mean} measure of each node. We use

a genetic algorithm to optimize the ratio of the scores. The genetic algorithm will be described in detail in Section 6.

4.5 Score Engine

Score engine will assign a final score to each suspect based on their geographical score and crime pattern score. The computation of crime pattern score is already described in Section 4, we will not describe that score in this section.

4.5.1 Geographical Score

Distance to crime scene is an important technique used in crime analysis. As discussed in Section 2, much recent work has focused on geographical profiling. One recent study shows that a logarithmic function works well in prioritizing the suspects of burglars [11]. Other studies [25, 26] show the mean distance of different types of crime travel to crime scene. We combine these methods to create a function to score the suspect based on their distance to the crime scene.

$$geoscore = 70 \times \frac{\ln(e + meandistance)}{\ln(e + distancetoscene)} \tag{6}$$

The geographical score *geoscore* is computed by Equation 6, in which *meandistance* is the distance derived from an empirical study on a certain type of crime [25, 26]. We use 1.62 miles as the *meandistance* for burglary and 2.1 miles as the *meandistance* for robbery. *distancetoscene* is the coordination distance between a suspect's living address and the current crime scene. e is the base of the natural logarithm to avoid the denominator of Equation being 0. Since our score system is based on a scale of 100, we use 70 to map the logarithmic values into the range 0-100. When the *geoscore* is greater than 100, we use 100 as the score.

4.5.2 Final Score

Final score of a suspect is a weighted combination of the three scores as shown in Equation 7.

$$finalscore = \frac{(w_1 \times geoscore + w_2 \times crimescore)}{100} \tag{7}$$

$$w_1 + w_2 = 100$$

In this equation, w_1 and w_2 are the weights associated with each score. They are computed by a genetic algorithm, which is presented in Section 6 to get the optimized configuration for final score.

4.6 Optimization Mechanism

As we mentioned in Sections 4 and 5, our score engine depends on an optimization mechanism to assign the optimized scores and weights. Our optimization mechanism is based on a genetic algorithm described in Figure 4.4. Our goal is to achieve the best median ranking of arrestees by applying the weights and scores to a sample data set. The sample data set contains 60 UCR reports from Anniston, AL police department. A UCR report contains the location of a crime, descriptions of the crime, and at least one arrestee of that crime. We run every report in FIRST and get a list of suspects ranked by their final score. Every returned list contains the arrestees for that crime. So, we can get all 60 arrestees' rankings in the returned lists and compute the median ranking of the 60 arrestees. The lower the median ranking is, the better the configuration of weights and scores is.

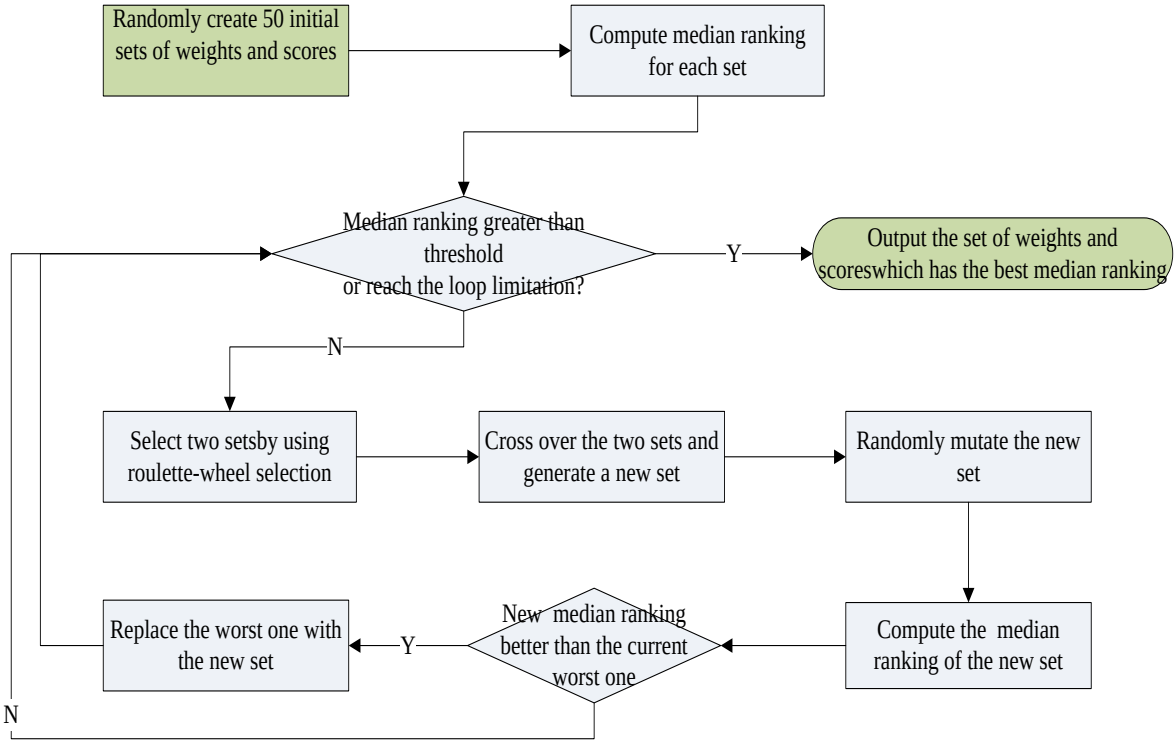


Figure 4.4. The genetic algorithm

The genetic algorithm will first randomly create 50 initial sets containing weights for the final score and for the scores of the decision tree. The seven rules in Section 4.2 are followed when creating scores in the decision tree. The algorithm computes the median ranking of each set of weights and scores and then starts the iterations of optimizing the weights and scores. We defined a median ranking of 25 as the threshold and a loop limitation of 10,000. The algorithm stops when either the threshold or the loop limitation is reached.

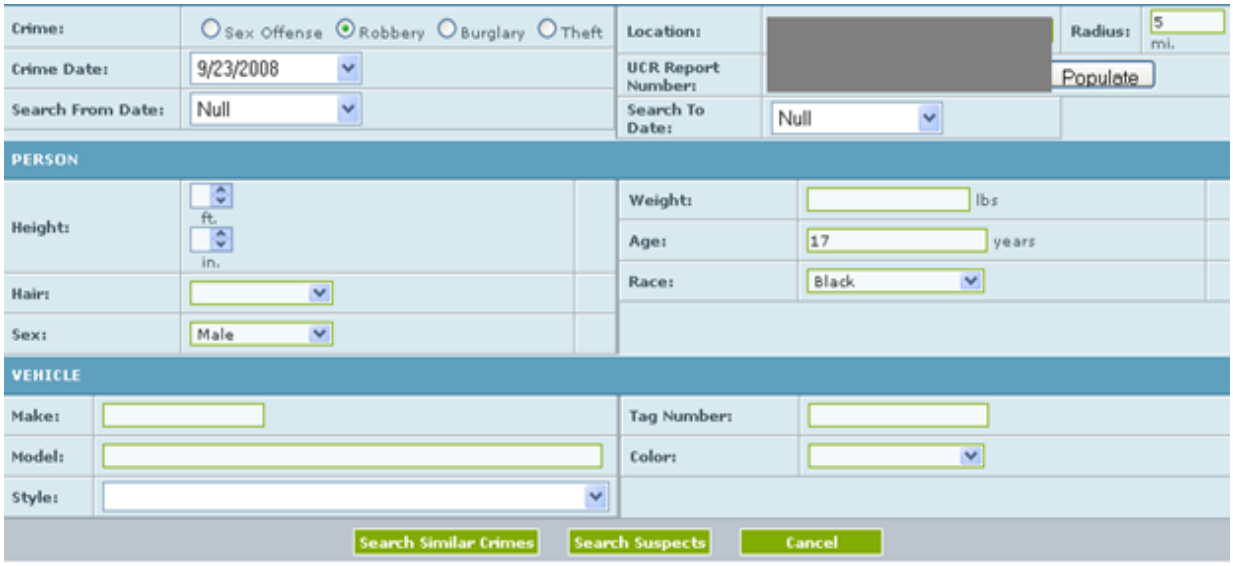
The core part of the genetic algorithm is creating offspring (a new set of weights and scores) from current inputs. In our system, we use the roulette-wheel selection method [27] to choose the two sets of weights to be the parents, to generate offspring. We then cross over the two sets by assigning 60 percent of gene from the parent having the best median ranking and 40 percent from the other parent to the offspring. After creating the offspring, we perform a

mutation on the offspring. To do this, we randomly choose two scores from the offspring set, and increase the first chose score by three points and decrease the second score by three points. Then, we compute the median ranking of the offspring set of weights and scores, if the median ranking is better than the current worst one, we replace that one by the offspring set. The genetic algorithm continues running until it reaches the threshold or the loop limitation. The set with best median ranking will be used in our system as the weights for computing the final score and the scores for the decision tree.

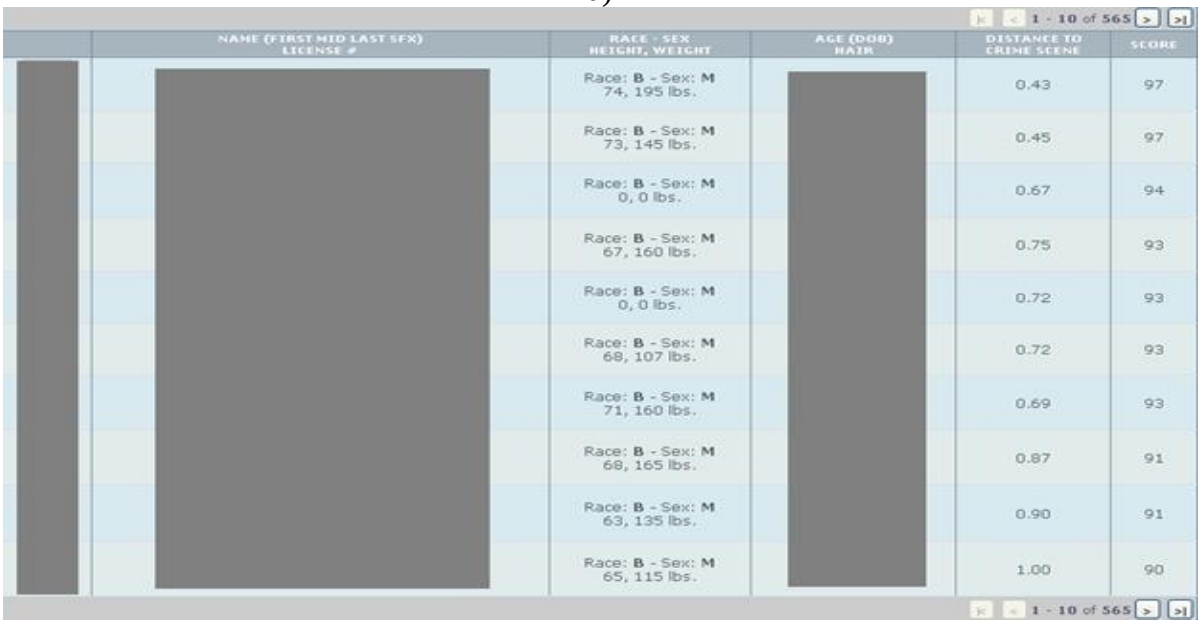
4.7 User Interfaces

Figure 4.5 shows the user interface of FIRST when running a real case in the system. We mask the sensitive fields to protect privacy. A police officer can enter the type and location of the crime. He/she can also enter the radius from the crime scene of interested. These fields are mandatorily required to run FIRST. The "PERSON" and "VEHICLE" sections are the physical descriptions of the suspect. They can either be manually entered or pulled out from an integrated UCR report system. They are not required by FIRST, but can improve accuracy.

After the officer completes the form, he/she can run FIRST with the specified criteria. FIRST will return a list of suspects ordered by the final score. In the results, we also include photos we obtain from our data sources. When the officer clicks on a suspect, FIRST will display the detail criminal history records and social network relationships of the suspect to help the officer better understand the background of the suspect.



a)



b)

Figure 4.5. The UI of FIRST

4.8 Experimental Results

To validate our approach, we use real burglary and robbery Uniform Incident/Offense Reports from Anniston, AL police department as inputs to FIRST and evaluate the performance of FIRST.

4.8.1 Experimental Design

We use 147 reports that each contains at least one arrestee so that we can verify whether FIRST returns the arrestee in the return list. Since the goal of the tool is to help police departments to identify the suspects, we did not take conviction into consideration. We first run all 147 reports in FIRST by only returning the suspects living within 12.88 miles and having at least one felony crime, without applying any scoring mechanism to them. We use a search radius of 12.88 miles because that is the longest distance from the center of Anniston to other adjacent cities. We then check the return list of FIRST to see if it contains the arrestee. 47 of the 147 reports don't return the arrestee. Of the 47 arrestees, 5 of them are living out of the 12.88 miles range, 10 of them do not have an address recorded in our database, and 32 of them do not have a previous felony crime. We use the 100 reports as our test bed to further analyze the performance of FIRST. There are 70 burglary and 30 robbery cases in the 100 reports.

Table 4.3. Parameters and Median Ranking

Parameters	Value
node1	100
node2	68
node3	56
node4	34
node5	17
node6	6
w_1	59
w_2	41
median ranking	244
median return list size	8846
median ranking in percentage	2.8

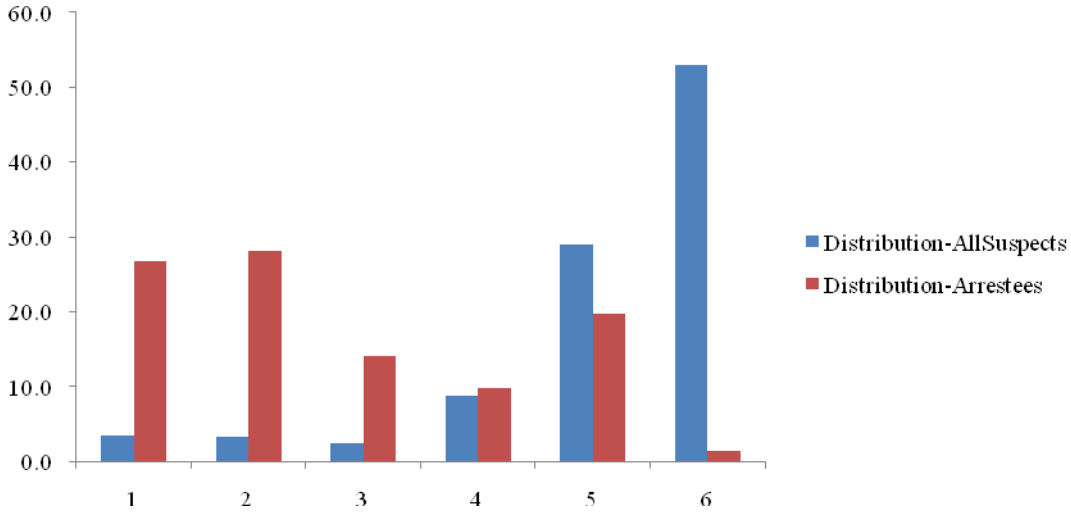


Figure 4.6. Distributions of different nodes among all returned suspects and arrestees

In our first experiment, we randomly choose 60 cases from the 100 reports as the sample data set to obtain the parameters of FIRST. Our random selection process guarantees that the portion of burglary and robbery cases remains the same as in the original 100 cases. There are 42 burglaries and 18 robberies in the sample data set. We then run the genetic algorithm mentioned in Section 6 to determine the weights of final score and scores of decision tree. The inputs of FIRST contain geo-coded locations of crime scene, crime occurred dates, and type of

crimes. The searching radius is 12.88 miles. After obtaining the optimized scores and weights, we apply them to the rest 40 cases as test data set to compute the median ranking of the test data set.

The first experiment does not consider the physical descriptions of the suspects. We design the second experiment to simulate the inputs of physical descriptions. Since we know the arrestees' physical description, we can trim the return lists from the first experiment by eliminating the offenders whose weights, heights and age are not in a reasonable range for the arrestees. In this simulation, we use age ± 10 years, weight ± 25 pounds, and height ± 3 inches as the filters. We then eliminate the suspects who are not in the given range and recompute the median ranking of the arrestees. We run this experiment against all 100 cases.

4.8.2 Experimental Results

Parameters. Table 4.3 shows the results of the optimization process. The median

Table 4.4. Median Rankings and Percentiles

Parameters	Overall Data	Sample Data	Test Data	Filter Data
Median	237	244	199	27
	10	14	8	1
Percentiles 10				
20	34	33	33	3
30	130	142	120	7
40	191	229	146	17
50	237	244	197	27
60	363	373	364	47
70	547	523	599	104
80	1393	1155	1466	208
90	2246	2092	2799	459
100	5887	5887	5509	3297

ranking of the arrestees in the 60 reports is 244. The median return list size of the reports is 8,846, which makes the median ranking on the top 2.8% of the return lists. Table 4.3 also shows the weights and scores the optimization process returned. The geographical score accounts for the large portion of the score (w1=59) and crime pattern score contributes 41 percent of the final score (w2=41).

We are also interested in analyzing the distribution of different nodes of the decision tree. Figure 4.6 is the comparison of the distribution of the nodes among the arrestees and among all the suspects returned within the return lists. From Figure 4.6 we can see that node1 has the least percentage among all the suspects but has the second highest percentage among the arrestees. Since node1 is the highest score of all the decision nodes, it will help us in prioritizing the suspects. We can also observe that node6 has the highest percentage among all suspects and has the least percentage among the arrestees, which will help us shift such offenders (offenders who have not commit felony crimes in last three years) to the bottom of the return lists.

The First Experiment. Table 4.4 shows the results when we apply the parameters to the 40 test cases. The median ranking of the test data set is 199. The median return list size is 8,659. The median ranking is on the top 2.3% of the return list. We also get the median ranking of the 100 cases which include sample and test cases. Table 4 also shows the median ranking of the whole data set is 237.

The Second Experiment. Table 4.4 also shows the results of the second experiment, in which we filter out suspects whose age, weight or height are not within a given range for the arrestees. Of the 100 cases, the median ranking is 27 and the median return list size is 804. This means the median ranking is still in the top 2.1% of the return list. In Table 4.4, we also include the percentiles of ranking of arrestees to help understand the performance of FIRST. We

can see from the results that the ranking of 30 percentile is seven, which means 30% of the arrestees are returned in the top 10 of the return lists. Our user interface displays 10 suspects on each page, which means the arrestee in on the first page of the return list. Our results demonstrate the promise of our approach in helping police officers to quickly identify suspects.

4.9 Conclusions

In this paper, we proposed a crime analysis framework for finding and prioritizing perpetrators. We build this framework in a pay-as-you-go fashion, which requires a minimum number of data sources to start. The required data sources like arrest records can be obtained by many states. When more data sources are available, they can be easily added to the framework. In this framework, we score each suspect by using their physical distances to the crime scene and a decision tree model of crime pattern. We use a genetic algorithm to improve the score of decision tree and the final score. Our experiments are based on real crime reports from Anniston, AL police department. The median ranking of the arrestees in FIRST is in the top 2.1% of the returned suspects list. We believe our approach to identifying perpetrators of a crime is helpful to law enforcements and can be adapted to other states.

References

[1] H. Chen, W. Chung, J. J. Xu, G. Wang, Y. Qin, and M. Chau, "Crime data mining: a general framework and some examples," IEEE Computer 2004, vol. 37, (4), pp. 50-56.

[2] S. V. Nath, "Crime Pattern Detection Using Data Mining," in Proc. IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology, 2006, pp. 41-44.

[3] J. J. Xu, and H. Chen, "Fighting organized crimes: using shortest-path algorithms to identify associations in criminal networks," Decision Support Systems, 2004, vol. 38, pp. 473-487.

[4] L. Ding, D. Steil, M. Hudnall, B. Dixon, R. Smith, D. Brown, and A. Parrish, "PerpSearch: An Integrated Crime Detection System," in Proc. IEEE Intelligence and Security Informatics 2009, Dallas, Texas, 2009, pp.

[5] http://caps.ua.edu/projects_lets.aspx

[6] L. Ding, and B. Dixon, "*Using an Edge-dual Graph and k-connectivity to Identify Strong Connections in Social Networks*," in Proc. ACM SE 2008, Auburn, AL, US, 2008, pp.

[7] D. Canter, "Geographical profiling of criminals," Medico-legal, 2004, vol. 72, (pt2), pp. 53-66.

[8] D. V. Canter, and A. Gregory, "Identifying the residential location of rapists," Journal of the Forensic Science Society, 1994, vol. 34, pp. 169-175.

[9] D. K. Rossmo, "Place, space and police investigations: Hunting serial violent criminals," 1995.

[10] L. H. David Canter, "A comparison of the efficacy of different decay functions in geographical profiling for a sample of US serial killers," Journal of Investigative Psychology and Offender Profiling, 2006, vol. 3, (2), pp. 91-103.

[11] D. Canter, and L. Hammond, "Prioritizing Burglars: Comparing the Effectiveness of Geographical Profiling Methods " Police Practice and Research, 2007, vol. 8, (4), pp. 371-384.

[12] R. V. Hauck, H. Atabakhsb, P. Ongvasith, H. Gupta, and H. Chen, "Using Coplink to analyze criminal-justice data," IEEE Computer, 2002, vol. 35, (3), pp. 30-37.

[13] D. J. Watts, and S. Strogatz, "Collective dynamics of 'small-world' networks," Nature, 1998, vol. 393, (6684), pp. 440-442.

[14] G. Kossinets, and D. J. Watts, "Empirical Analysis of an Evolving Social Network," Science, 2006, vol. 311, (5757), pp. 88-90.

[15] L. Backstorm, D. huttenlocher, J. Kleinberg, and X. Lan, "Group Formation in Large Social Networks: Membership, Growth and Evolution," in Proc. 12th ACM SIGKDD, Philadelphia, PA, USA, 2006, pp. 44-54.

[16] S. Kaza, D. Hu, and H. Chen, "Dynamic Social Network Analysis of a Dark Network: Identifying Significant Facilitators," in Proc. IEEE Intelligence and Security Informatics, New Brunswick, NJ, USA, 2007, pp. 40-46.

[17] S. R. Safavian, and D. Landgrebe, "A survey of decision tree classifier methodology," IEEE Trans. Systems Man Cybernet, 1991, vol. 21, pp. 660-674.

[18] H. Chen, H. Atabakhsh, C. Tseng, B. Marshall, S. Kaza, S. Eggers, H. Gowda, A. Shah, T. Petersen, and C. Violette, "Visualization in law enforcement," in Proc. Human Factors in Computing Systems, Portland, OR, USA, 2001, pp. 1268 - 1271.

[19] www.esri.com

[20] N. Levine, "CrimeStat: A Spatial Statistics Program for the Analysis of Crime Incident Locations (v 3.1)" (Ned Levine & Associates, Houston, TX, and the National Institute of Justice, 2007. 2007).

[21] B. Ostrom, M. Kleiman, F. Chessman, R. Hansen, and N. Kauder, "Offender Risk Assessment in Virginia: A Three-Stage Evaluation," 2002.

[22] R. Barnoski, and S. Aos, "Washington's Offender Accountability Act: An Analysis of the Department of Corrections' Risk Assessment," 2003.

[23] http://www.ojp.usdoj.gov/bjs/abstract/rpr94.htm

[24] I. H. Witten, and E. Frank, "Data Mining: Practical machine learning tools and techniques" (Morgan Kaufmann, 2005, 2nd edn. 2005).

[25] M. Laukkanen, and P. Santtila, "Predicting the residential location of a serial commercial robber," Forensic Science International, 2005, vol. 157, (1), pp. 71-82.

[26] B. Snook, "Individual differences in distance travelled by serial burglars," Journal of Investigative Psychology and Offender Profiling, 2004, vol. 1, (1), pp. 53-66.

[27] D. Whitley, "A genetic algorithm tutorial," Statistics and Computing, 1994, vol. 4, (2), pp. 65-85.

CHAPTER 5

CONCLUSIONS

We have presented a framework that utilizes data mining techniques, social network analysis and geographical profiling to assist crime investigation and predicting recidivism. In the framework, our proposed risk assessment model has shown strong correlation to future offense; our novel method to measure strong ties in social networks has been proved to be robust against noisy data. Also, the social network we constructed supplied fundamental data that are used to predict recidivism and suspects in crime investigations; the case study on identifying perpetrators is using real police reports and the results show that our framework can help the investigator to narrow down the searching list. In particular, the decision tree model of creating crime pattern combined with geographical profiling show the strength of predicting where and who the suspect is.

We believe the research questions addressed in this dissertation can be extended to other domains as well. For example, the measurement of strong ties in social networks can also be applied to other type of social networks: online community networks, author-coauthor networks etc.

The future direction of this research work will be analyzing more types of offenses and creating more accurate crime pattern based on more real police reports. We also plan to extend the social network analysis to above mentioned areas.

REFERENCES

[1] H. Chen, W. Chung, J. J. Xu, G. Wang, Y. Qin, and M. Chau, "Crime data mining: a general framework and some examples," IEEE Computer 2004, vol. 37, (4), pp. 50-56.

[2] S. V. Nath, "Crime Pattern Detection Using Data Mining," in Proc. IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology, 2006, pp. 41-44.

[3] J. J. Xu, and H. Chen, "Fighting organized crimes: using shortest-path algorithms to identify associations in criminal networks," Decision Support Systems, 2004, vol. 38, pp. 473-487.

[4] D. V. Canter, and A. Gregory, "Identifying the residential location of rapists," Journal of the Forensic Science Society, 1994, vol. 34, pp. 169-175.

[5] D. K. Rossmo, "Place, space and police investigations: Hunting serial violent criminals," in Proc. Conference Name, Conference Location, 1995, pp. 217-235.

[6] D. Canter, "Geographical profiling of criminals," Medico-legal, 2004, vol. 72, (pt2), pp. 53-66.

[7] D. Andrews, "Recidivism is predictable and can be influenced: using risk assessments to reduce recidivism," in Proc. Forum on Corrections Research, 1989, pp. 11-18.

[8] D. Andrews, and J. Bonta, "The Level of Service Inventory-Revised," in Proc. Conference Name, Conference Location, 1995, pp.

[9] B. Ostrom, M. Kleiman, F. Chessman, R. Hansen, and N. Kauder, "Offender Risk Assessment in Virginia: A Three-Stage Evaluation," in Proc. Conference Name, Conference Location, 2002, pp.

[10] J. Austin, D. Coleman, J. Peyton, and K. Johnson, "Reliability and Validity Study of the LSI-R Risk Assessment Instrument," in Proc. Conference Name, Conference Location, 2003, pp.

[11] R. Barnoski, and S. Aos, "Washington's Offender Accountability Act: An Analysis of the Department of Corrections' Risk Assessment," in Proc. Conference Name, Conference Location, 2003, pp.

[12] R. Hare, "The Psychopathy Checklist-Revised, 2nd Edition" (Toronto: Multi-Health Systems, 2003. 2003).

[13] M. S. Granovetter, "The strength of weak ties," American Journal of Sociology, 1973, vol. 78, (6), pp. 1360-1380.

[14] X. L. Shi, L. A. Adamic, and M. Strauss, "Networks of Strong Ties," Physica A: Statistical Mechanics and its Applications, 2007, vol. 378, (1), pp. 33-47.

[15] J. Clark, "A first look at graph theory" (World Scientific, 1991. 1991).

[16] F. Harary, "Graph Theory" (Addison-Wesley, 1969. 1969).