8-2022

Generating Sex Trafficking Networks From Text Documents

Maria Diaz
mddiaz@g.clemson.edu

Generating Sex Trafficking Networks From Text Documents

A Master's Thesis
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in
Industrial Engineering

by
Maria Diaz
August 2022

Accepted by:
Dr. Thomas Sharkey, Committee Chair
Dr. Yongja Song
Dr. Emily Tucker

Abstract

Qualitative coding is a long and strenuous process that requires a well-skilled investigator. Natural language processing techniques have made leaps and bounds as far as usability and application domain, although it does not work for every task. In this work, we have created a natural language processing framework to help qualitative coders automatically obtain the nodes and node arcs from federal case files, dockets, and indictments within a sex trafficking network. The produced nodes and arcs allows us to perform network modeling by providing us with the information needed to create network structures that can then be used for interdiction simulation. The network models can also be analyzed for patterns, trends, and contrasts. Another goal for these networks is to apply Operations Research (OR) methods to better understand the operations of sex trafficking networks. Results fared better for the node extraction task, begging the question, does automation belong in the process of coding sex trafficking networks? If yes, then future implementations should avoid rule-based matching, despite the high structure of court documents. Additionally, more data would help improve accuracy of a model; however, obtaining ground truth data requires human coders. This thesis helps to address the question of how automated techniques, such as natural language processing and machine learning, can play a role in qualitative coding and thematic analysis. Further, by focusing on obtaining networks from text documents, it provides a basis for inputs into operations research models.

Dedication

To Ricardo: My love, my light, my future, my might.

Acknowledgments

Tom Sharkey (Clemson), thank you for being my advisor. You have helped me achieve so much in my short amount of time here. Your knowledge and creative eye were instrumental to this work.

Kayse Maass (Northeastern), Lauren Martin (University of Minnesota), Christina Melander (RTI), Kelle Barrick (RTI), thank you all for sharing your resources with us. This project would not be possible without the data you provided, the coded spread sheets for reference, and the gentle encouragement.

Emily Singerhouse (Strategic Prevention Solutions), thank you so much for showing me how to use and understand the codebook. This project could not have been completed without your patience and expertise.

Emily Tucker (Clemson), Yonga Song (Clemson), thank you for agreeing to be on my committee. I greatly appreciate the extra time you two have spent looking through my thesis.

To the many others I may have missed in this section, thank you.

We recognize that our research cannot capture all the complexities of the lived experiences of trafficking victims and survivors. The trafficking victims and survivors are not just data, they are people who have experienced a great trauma. It is our ambition to develop tools to help us understand and strategize on sex trafficking networks, however, we want to minimize any distress as much as possible while working toward our goals. We take this space to make this acknowledgement and express our respect to the trafficking victims and survivors.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

By United States law, human trafficking is defined as the use of force, fraud, or coercion to obtain labor or commercial sex acts from a victim against their will. Sex trafficking exclusively refers to the side of this that accounts for commercial sex acts. Efforts have been made to acquire metrics and statistics from persons currently in or out of sex trafficking networks [2] but the difficulty in finding quality and consistent data in the face of intentional concealment, moral ambiguity, and societal taboos proves increasingly difficult. There is also the issue of re-traumatizing victims in order to retrieve their personal account of the events as well as a risk of triggering relapse [3]. Furthermore, inconsistent data collection methods increase the challenges of performing cross longitudinal studies and using similar analytical methods. In order to circumvent these issues, researchers are finding new ways to analyze existing data or identifying practices and limitations of extracting key data from available sources in order to better understand sex trafficking networks.

1.0.1 Background

Sex trafficking involves the use of force, fraud, or coercion to induce a commercial sex act from an adult victim [4], or involves the facilitation of a commercial sex act from minor victims, (i.e. no force, fraud, or coercion is necessary since a minor cannot give consent) [5]. According to the nonprofit Trafficking in America Task Force, sex trafficking accounted for 99 billion of the 150 billion USD total made from human trafficking worldwide in 2020 [6]. This number is likely a gross underestimation as trafficking often massively exceeds what information is reported. Researchers have found that for two study sites, with populations around 600K and 2.3 million, human trafficking events recorded in law enforcement and social service agency records most likely accounted for only 14 to 18 percent of the total potential trafficking victims [6]. Under-reporting resulting in under-counting is often caused by persistent issues with local trafficking identification and incident reporting [7]. Therefore, the sex trafficking problem is far larger than what is known from the current data. Additionally, this contributes to the sparsity of data sets and decreases their predictive accuracy. Certain data focuses on how commercial sex is advertised, which essentially is its ‘public facing’ piece; case file analysis is limited by the types of sex trafficking operations that are prosecuted (which may not be representative of the landscape of trafficking).

Despite these issues, there have been many academic pursuits aimed at gaining insight and awareness into the world of sex trafficking. One such pursuit involves social network analysis. Social network analysis studies the behavior of individuals within the network, patterns of relationships, and dynamic structure of the network itself. Analysis can identify clustering, centrality, and superimposable qualities of the structure as well as the ability to provide direct comparison to other networks [8]. In operations research, supply networks can be used to assist in optimal resource allocation or even network interdiction [9]. In anti-sex trafficking efforts, network analysis has been used to generate victim networks [5], generate synthetic sex trafficking network models for interdiction modeling [1], identify ads that interconnect with each other in similarity as a tool for law enforcement [10], and investigate child sex trafficking networks by way of social network analysis [5]. Network analysis and modeling is certainly an effective tool and has a place in anti-sex trafficking efforts. Network analysis provides a quantitative way to analyze data, whatever the purpose.

Qualitative analysis allows for a different type of result, but similarly to quantitative analysis, it allows us to gain insight and awareness into the topic of sex trafficking. Qualitative research is employed when the individual would like to gain a deeper understanding of a phenomenon. It also allows the individual to identify existing patterns from data at a micro and macro level [8]. This kind of analysis can give researchers a look at cause and effect, features of interest, and possible solutions to the problem [11]. Qualitative analysis has many categories of methods to analyze data, however, we focus on only one such method known as thematic analysis, and under this umbrella more specifically, qualitative coding. Qualitative coding is useful to researchers, but it takes a highly skilled qualitative researcher at great cost to their own time to perform as there exists a massive amount of text data and a very finite number of qualitative coders.

Textual data for the purpose of qualitative coding can be found in many forms. Qualitative coders can analyze interviews, surveys, and archives. When it pertains to sex trafficking, qualitative coders can analyze victim statements, police reports, court documents, and even federally prosecuted case files. Because of the hidden nature of sex trafficking data, the type of publicly available information research teams can get is limited [12]. This is one of the main reasons why our qualitative coding partners chose to analyze federally prosecuted case files. Unfortunately, our own qualitative research partners experienced the same difficulties with qualitative coding: Too much text, not enough time, therefore, in this work, we introduce a way to automate the node and relation extraction portion of the process which we will compare with the human-coded versions in our analysis. This will help us determine the accuracy of using automated methods in node and relation extraction tasks and to what degree of automation we can reasonably expect.

1.0.2 Objective

We may be able to better understand the operations of sex trafficking (which we will sometimes refer to as ST) through network analytics. For example, Cockbain [13] applies in depth social network analysis to 6 prosecuted sex trafficking networks in the United Kingdom. She developed a qualitative codebook to identify nodes and arcs within each sex trafficking network. However, qualitative coding can be time-consuming and the goal of this thesis is to explore natural language processing methods to determine whether they can help construct sex trafficking networks from case file data and find where the limitations of these techniques exist. Our work includes automatically generating networks from the resulting spreadsheets and comparing them with human-coded networks to measure performance (See Figure 1.2 for a sample network comparison). We have unique access to a qualitative codebook that was developed by domain experts in sex trafficking and operations researchers, that was applied to data surrounding 13 federally prosecuted cases by qualitative researchers.

Using the case file data, we key in on two main tasks: Node extraction and relation or “arc extraction”. The purpose of node extraction is to extract all of the main entities in the trafficking network and assign the correct “code” or label. The label describes what the extracted item is, including whether they are a trafficker, a hotel, or a victim. Next, this list of key entities and their codes are exported as an excel spreadsheet known as “node data”. Then we move on to perform relation extraction. Here our goal is to extract the relation between two main entities in the network.

This will automatically generate the “relationship arc” spreadsheet thus completing the automation portion of our work. We can then use operations research (OR) methods to model the operations and, potentially, disruptions to sex trafficking networks. We can also model a network using the node data and relation arc data and compare the result to human coded data to measure the utility of our results and determine limitations.

In some approaches, the data problem is addressed by re-purposing previously used data. Dubrawski et al. [14] used previously collected escort ad data, sex trafficking surveys, and temporal data of posted commercial sex advertisements to extract new data analysis and create new anti-sex trafficking tools. Their own analysis yielded a predictive classifier to identify advertisements that possibly express trafficking activity and an entity resolution method to document and follow potential advertised victims. Szekely et al. [10] used web crawling to extract advertisement information from existing websites. Their results were used to produce knowledge graphs of victims, ads, and phone numbers. Our research team reused existing case files generously provided to us by the University of Minnesota, along with the networks that were created by applying a codebook to identify nodes and arcs. Our own analysis did not rely on any data collection within the study.

As previously mentioned, one way that data is analyzed is through qualitative analysis, specifically qualitative coding. Qualitative analysis is an umbrella term used to describe a method of coming to understand a complex and nuanced way of comprehending a phenomenon given some kind of data [11]. Thematic analysis falls under the category of qualitative analysis and qualitative coding falls under the umbrella of thematic analysis. Qualitative coding is described as prescribing a set of codes to assign patterns, behaviors, and summaries to a piece of information [11]. This process is essential to qualitative analysis because it can help researchers uncover trends and learn the specificalities of the data. Qualitative coding can also be quite time consuming, but especially so when the data is large or heavy in volume [15]. In this study, our data is comprised of federally prosecuted case files, which include the court’s charges against an individual as well as an account of the criminal event [16]. We will also use dockets and indictments which contain penal codes and charges. Nodes and relationship arcs are extracted from the case files, indictments, and dockets by qualitative coders and saved in a series of excel spreadsheets. Our study defines nodes as an entity in the recounted case file, docket, or indictment text and an arc is a description of the relationship between them. The spreadsheets are used to populate network models and graphs to aid in efforts that interrupt sex trafficking.

We would also like to use these networks to apply Operations Research (OR) methods to help us better understand the operations of sex trafficking networks. Through this understanding, we can better assist victims, interdict supply chains, direct resources, and so much more.

1.1 Methods Overview

Automation offers additional help to qualitative coders and may speed up the processing time and improve the volume of work produced, resulting in larger amounts of data being processed. In our own work, our human-qualitative coder approximates that it took over 50 hours to code 13 networks [17]. There have been efforts to automate some or all of the qualitative coding process through Natural Language Processing (NLP) [18] [19] [20] [21] [22]. Natural language processing refers to the science of using statistical and computational modeling of the nuances and qualities of language and developing processes relating to language [23]. In our study, a series of NLP tools are employed including dependency parsing and rule-based matching. We also employed convolutional neural networks via SpaCy Named Entity Recognizer.

1.1.0.1 Dependency Parsing

Dependency grammars can be described as two-way directed dependencies, or relations, between two words [24]. Dependency grammars are visualized as a typed dependency structure and typically fall into two camps: transition-based parsers and graph-based parsers. Our dependency parser is a graph-based one. The terms *head* and *child* are used to represent two words connected together by a single arc [25]. Figure 1.1 shows an arc which symbolizes a syntactic relationship that joins the child to the head node. A sentence often has words with many syntactic relations with many arcs visualized over the sentence. When this happens, this visualization is known as a dependency tree which is also expressed in the figure. In our work, we parse the dependency tree to look for specific relations that indicate specific events in the text. We take these dependency patterns and create rules from them. These rules are used to seek out every instance of the syntactic pattern and isolate for further data manipulation.

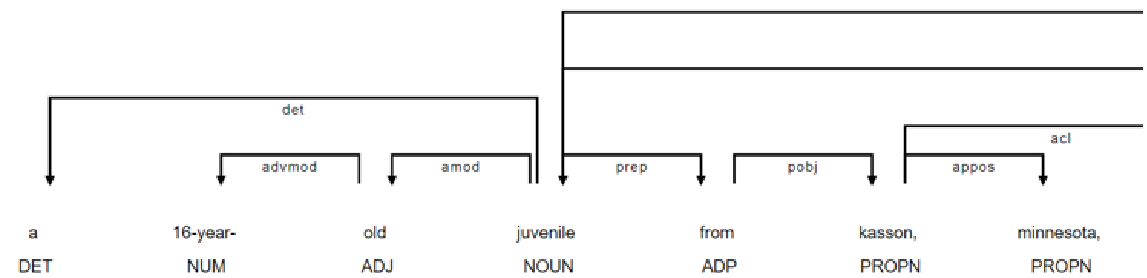


Figure 1.1: A dependency tree with nine branches. Dependency trees help us find patterns to make rules from.

1.1.0.2 Relation Extraction

Relation extraction is a sub-task defined as extracting structured semantic relationships from an unstructured source, usually text [26]. It is extracting facts between two entities from a piece of text and is often pursued as an automated process. The objective of relation extraction is to take unstructured data and transform it into structured text that describes the interdependence between two or more named entities [27]. There are three relation extraction methods to create networks from. They are unsupervised methods, supervised methods, and knowledge-based methods. For the purpose of this research, we focus on knowledge-based methods. Our data is domain specific, has a close set of relations that need to be extracted, and similar patterned text, making knowledge-based relation extraction a viable option. Additionally with the limited amount of data our team had access to, using supervised or unsupervised methods would not be a suitable choice.

1.1.0.3 Convolutional Neural Network

A convolutional neural network, or CNN, is a type of artificial neural network. An artificial neural network is a category of computational processing systems that collectively learn from a given input to optimise it's final output. A CNN is a machine learning tool that loads user input as a vector to a an input's hidden layers. The hidden layers make choices based on the conclusions of the previous layer, checking stochastic differences within and whether it worsens or improves the resulting output [28] (See Figure 1.3 for more details). SpaCy's named entity recognizer, which we use in this research, uses a CNN-type structure to perform Named Entity Recognition [29].

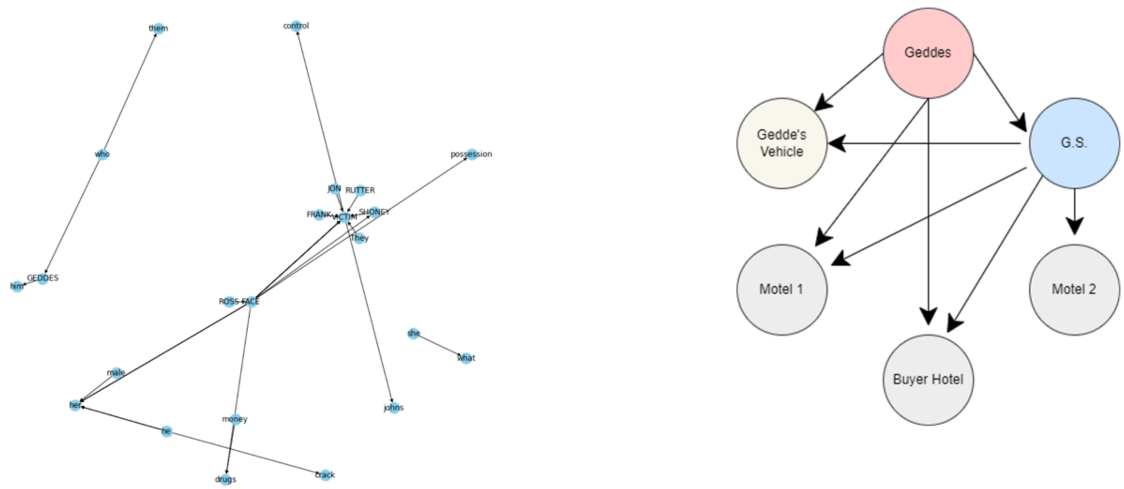


Figure 1.2: An idea of what the computer-generated networks look like versus a human-coded one. Notice that the computer-generated network contains more nodes with extra details while the human-coded network includes a more deliberate outcome.

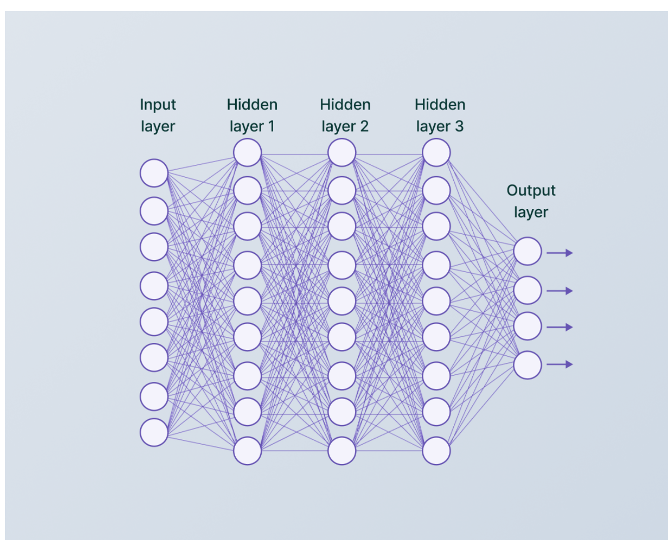


Figure 1.3: A CNN's process. Each hidden layer is one convolution, abstracting the input until the final output layer.

1.1.0.4 Named Entity Recognition

A named entity represents an object that has a proper name such as a state named Hawaii or a person named Peggy [29]. The named entity recognizer, or NER, uses machine learning to predict the many names of entities in a given text data. SpaCy's NER uses a pretrained ML model to perform the categorization task. SpaCy's NER uses a custom deep learning network that is based on the principles of a CNN. This can be difficult for our case files since victim names may be redacted in publicly available case files, causing redacted entities to remain hidden and entries to be left out of the resulting node data spreadsheet.

1.1.1 System Overview

The unique benefit of leveraging case file data, dockets, and indictments, as demonstrated in our study, is that it has the potential to increase the volume of workable information in the current domain. The original case files, dockets, and indictments data comes to us in the form of scanned PDFs which include pages with notable imperfections. The documents are converted to images before being converted to text. From here, the text is cleaned and training sentences are extracted using nodes as training words and labels defined by the qualitative coders. The data is used to add a Named Entity Recognition (NER) pipe to the existing NLP model. Next, the model goes through a validation step. Validation in machine learning is an essential step in estimating the general performance and predictive power of the model on untested data [30]. This model is then applied to the data and used to extract nodes and assign the codes to the text. Then, relation extraction is performed using rule-based matching and language rules we have observed to be effective in elucidating the final results. Please see Figure 1.4 for an illustration of the framework overview.

1.1.2 Contributions

As previously mentioned, our approach has the potential to increase the amount of information to a domain with limited and restricted data. While other studies have focused on applying techniques to other modalities of information, our approach leverages currently existing data and transforms it into the final network models. The resulting code will be placed on GitHub where it can be tried out on other case files from other research groups. In addition to this, we would

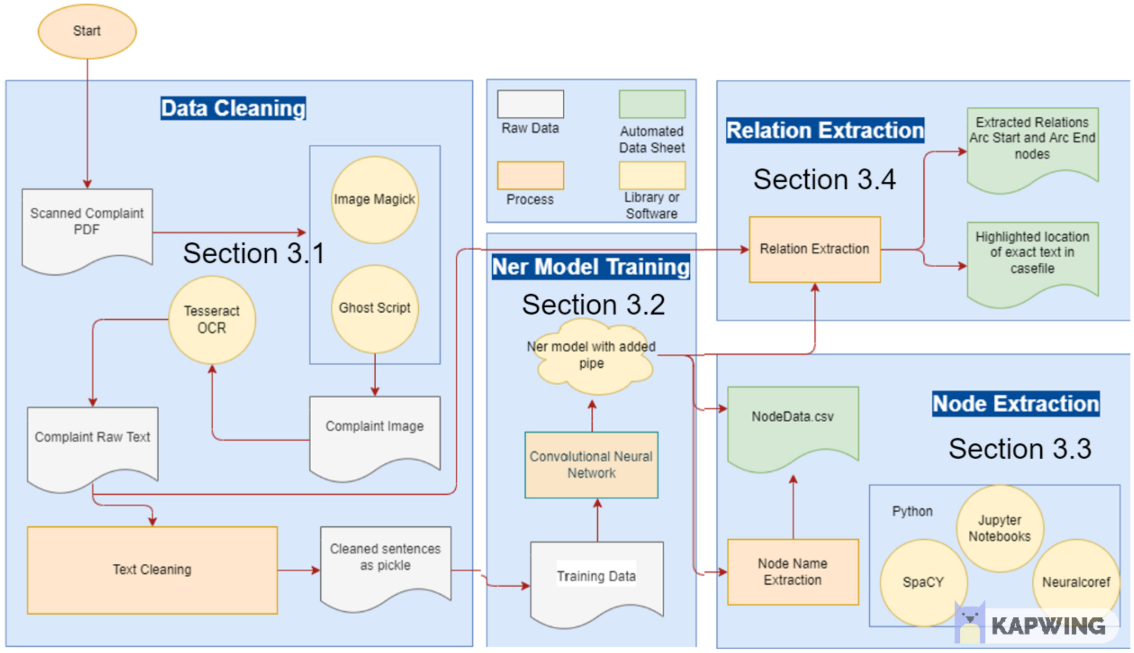


Figure 1.4: Overview sectioned into three distinct parts with chapters pertaining to each section

also like to explore limitations of using automation of processing case files to identify networks. In doing so, we would like to identify what those limitations are and possible remedies. This thesis helps to address the question of how automated techniques, such as natural language processing and machine learning, can play a role in qualitative coding and thematic analysis. Further, by focusing on obtaining networks from text documents, it provides a basis for inputs into operations research models.

1.1.3 Thesis Organization

The rest of this thesis is organized into 5 parts. Chapter 2 examines related work and reviews current NLP techniques, automation, ML, and anti-sex trafficking. Chapter 3 discusses the methods of this study and goes into detail about data, techniques, and existing biases within our data/method. Chapter 4 reports the analysis and results. Chapter 5 discusses important takeaways from this work and summarizes the next steps for work in this area.

Chapter 2

Related Work

In this section, we would like to provide background regarding the topics illustrated in the coming chapters. First, we discuss Natural Language Processing (NLP) and the current advancements it has made in qualitative coding. Next is a short summary of NLP techniques used in automation followed by a section about machine learning used within relation extraction tasks. The chapter is concluded with work that has been done in the anti-sex trafficking domain.

2.0.1 NLP and Qualitative Coding

Natural Language Processing, or NLP, is increasingly being innovated upon and applied to the task of automatically coding qualitative data. As in our work, Crowston et al. [20] used a manual rule-based approach as well as machine learning to provide semiautomatic qualitative coding, alternatively, their machine learning approach used the features known as bag of words, location only, and parts of speech tagging. Our approach uses pre-labeled data to train a convolutional neural network pipe and our rule-based matcher uses rule-based pattern matching with the SpaCy module. In their work, Crowston used regular expressions, lexicon analysis, and other linguistic phenomena that exist within the text to create coding rules while our methods involve rule-based pattern matching and machine learning to extract labeled nodes. They also used a machine learning approach to learn the patterns of the extraction decisions by leveraging statistical and semantic feature techniques to the text, similar to SpaCy dependency trees.

Marathe et al. [22] engineered prototypes to partly automate the qualitative coding process using NLP techniques. Their strategy is focused on qualitatively coding phrases included in the text and using search style query-matching techniques to do so. Our own methods used SpaCy named-entity recognition to identify labeled nodes from pre-labeled data. Leeson et al. [21] used the tools Word2Vec and Topic Modeling on their text data to assign codes to unlabeled data. Our own approach uses training with pre-labeled data to assign codes to new data.

In Patton et al. [19], social work researchers collaborated with data scientists to use twitter data and qualitative analysis to predict potentially violent future interactions from tweets. In our own work, we use scanned PDFs of case files, dockets, and indictments to predict labels and extract relationship arcs. The federally prosecuted files are more structured than twitter tweets, so it made more sense to include a rule-based approach to relation extraction. Furthermore, our project is less of a predictive tool in terms of future actions but in a way produces a predicted result.

There have also been many advancements in the use of Natural Language Processing in qualitative analysis overall. Abram et al. [15] developed an assistive NLP tool for use with a human qualitative coder. In their work, they used the data from nine confidential interviews with nurses at a substance abuse facility which had been previously analyzed. Results proved promising, reducing work costs by 1500 dollars and reducing project time by 120 hours. Another benefit to their work is the creation of a domain specific corpus. Having a customized experience offers easier to interpret results. Large volumes of text can be systematized, networks can be etched out and analyzed, sentiment can be determined, and documents can be coded by way of machine learning prediction [31]. Qualitative analysis and natural language processing have the potential to unlock hidden knowledge and patterns inaccessible with human-only methods as well as produce a far greater amount of coded text results in a shorter amount of time; however, this should likely be a continued collaborative effort between qualitative coders and NLP as domain experts will always be needed because codebooks are often necessary and would be best done by the domain experts.

2.0.2 Automation and Anti-Trafficking Efforts

Our own use of natural language processing methods to combat sex trafficking are not unique. In fact, NLP tools are being regularly built and used to treat data sources found on the internet with the intention of fighting against ST.

Alvari et al. [32] employ a website to hone in on patterns of human trafficking-related online activities, including advertisements, which are immediately useful and passed on to law enforcement.

Similar to our own work, they used human labeled data to create a training set; in contrast, we use data from federally prosecuted case files, dockets, and indictments which are known to include information about criminal acts and use this to create our training set.

In Tong et al. [33], researchers build Human Trafficking Deep Networks, or, HTDN that automatically identifies trafficking advertisements. Their work uses text and images, two separate modalities of information. Alternatively, in our approach, we utilize text data alone.

Mensikova et al. [34] integrate multiple sentiment analysis algorithms and apply these to text in online advertisements crawled from the open web. Within their work, developers integrated several sentiment analysis algorithms and used these to treat the text of web-crawled advertisements found online. In contrast to our own work, we use dependency parsing and a machine learning model to extract nodes and arcs for graph models and our data set is not available to the public on the open web.

Wang et al. [35] built a system that includes NLP abilities, known as TrafficBot. TrafficBot is described as a data warehouse that uses two automated task assistants, information retrieval and integration, to gather data for law enforcement. The project uses alias detection, extraction, canonicalization, and cross-source correlation to scan escort and massage services from open sources with fewer resources needed.

Hultgren et al. [36] suggest more research on how to automate a system for identifying third party speech, age, and alias inconsistencies. They also suggest using a dynamic keyword ontology while using a knowledge management approach to continuously update the keyword ontology using data from rescued trafficking victims. This approach allows for consenting sex workers to be left out of police targeting while helping individuals being trafficked to be removed from the trafficking network. They also suggest scraping ads for more keywords on a regular basis to find more influential words to add to the ontology.

Diaz and Panangadan [37] developed an automated way to locate illegal massage businesses by sifting through Yelp reviews. Their novel approach resulted in a binary classifier that predicts whether or not a massage business on Yelp is actually an illicit massage parlor. The methods include the building of a data set and processing the data into a Document-Term matrix for training a binary classifier. The data set included the Yelp academic dataset and a set of data from a known illegal massage business verifying website to verify what labels to use on which reviews.

2.0.3 Machine Learning in Relation Extraction

On the topic of automated tasks, through natural language processing techniques, the process of identifying equivalent information between data sets or databases has been exhaustively researched for relational databases. This practice is known as *Record Linkage*. For such data sets, the task becomes *relation extraction* – identifying semantically related artifacts within a set, almost exclusively with text data. Relation extraction is based on extracting semantic relationships between the entities from text. [38]. The resulting data can be used to take raw text to highly organized, structured text that can be used to perform other tasks. Typically, the process is performed as two steps, in which the entities are extracted using a Named Entity Recognizer (NER) before relation classification is used to identify any pair-type relationship between entities.

Document-level relation extraction is another ambition in relation extraction, specifically with innovations that tackle barriers to relation extraction accuracy. In Tan et al. [39], researchers crafted a three way technique for document extraction. Researchers used an axial attention module for learning the inter-dependencies between entity-pairs and employed adaptive focal loss to treat class imbalance. They also used knowledge distillation to remedy the inconsistencies between human annotated data and supervised data. Results are promising and the work boasts an F1 score of 67.28.

Xu et al. [40] have that dependencies are modeled and used with document-level relation extraction. The researchers developed their own system, known as SSAN, in which the proposed model receives a text input and constructs contextual representations using entity structure within itself and throughout the overall encoding process. Through their two transformation designs, they achieve structural reasoning and contextual reasoning that exceed the competitive baselines.

2.0.4 Anti-Sex Trafficking: Efforts, Aspirations, and Predictions

There is a landscape of opportunities for researchers to aid in the fight against ST. Operations research offers several ways to remedy the situation including modeling supply chains, big data analytics, resource allocation, network disruption, decision making, and interdisciplinary efforts [9], but the key to any work in this area is to apply techniques to properly constructed, and realistic, data.

In Xie and Aros-Vera [41], ST networks were captured for the purpose of creating an interdependent network interdiction model that solved itself based on duality theory between the information gleaned and the physical victim. The model maximizes the ability of the interdiction by choosing when to gather information and when to arrest traffickers. Results suggest optimality in creating a case prior to trafficker arrest and using defensive measures to maximize the impact of interdiction. Similar to our work, Xie and Aros-Vera [41] used federally prosecuted case files.

Recall that data is exceptionally limited in the ST domain. In Figure 2.1, the top part of the pyramid represents data that is available to the public. The next level represents agency-collected data, like our own case files, dockets, and indictments used in this research. Finally, at the bottom of the pyramid are hidden data. Most of the data researchers have access to are in the top two tiers, however, there is an inaccessible abundance of data hidden or intentionally abstracted from the public as is the nature of illicit businesses. There are researchers who aim to bring more of that hidden data to the surface, so that a clearer picture of ST networks can be accessed.

Kosmas et al. [1] produces synthetic operational and social connections among persons experiencing trafficking, bottoms (which are victims that are forced to traffic others), and traffickers. This research allows operations researchers and trafficking teams to have a source of data without having to find and clean or construct it on their own. This should increase engagement between researchers and the trafficking problem. Kosmas et al. was also able to apply network interdiction to the networks created by their proposed network generator. Uniquely, the research team honed in on a way to model network flow that takes the ability of traffickers to control victims into account. This work addresses both the problem of limited ST data and provides a novel way to include a new dimension to modeling the network flow of this problem.

Keskin et al. [42] combines operations research and information systems concepts to find clusters of posts and predict movements based on text, images, and phone numbers. Their research also assists other research teams to determine which data are available when constructing advanced interdiction models. This result gives law enforcement the opportunity to help identify ST organizations as well as individuals, furthermore, this result is highly desirable as current law enforcement methods only identify information tied to a specific suspect while ignoring high-level patterns within the ads they are tracking. Unique to their study is the sheer volume of their data set which contains 10 million advertisements. This has the potential to increase the reliability and consistency of any predictable result.

Finding where and how ST is occurring is no easy task. Geo-spatial data on this topic is in short supply and data gathering methods are insufficient. Difficulties include coordinating several

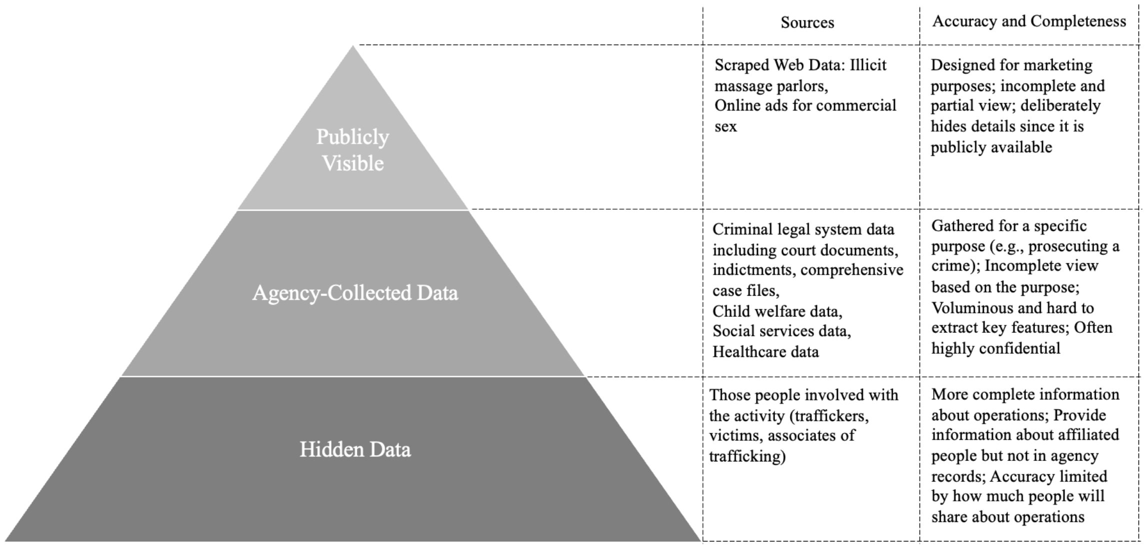


Figure 2.1: This pyramid first presented by Kosmas et al. [1] illustrates ST data accessibility. Federally prosecuted case files fall into the middle part of the pyramid.

geographies at the same time, uncertainty in geographical locations, disaggregation and diversity, and uncertain paths [43].

Simonson [44] used semi-supervised machine learning to automate a binary classifier that determines if a post is related to ST or not. Through their work, a fully labeled data set can be used to identify locations where resources and sponsorships should be directed to people experiencing trafficking by location analysis, as the tool can help answer when and where ST is occurring. Additionally, Simonson [44] was able to determine that the FOSTA/SESTSA laws do not work as intended, as they cause social media companies to target individuals selling sex instead of the traffickers. Once the individual selling sex is removed from the platform, they have a harder time vetting customers on the street who may be dangerous.

Tripp [45] agrees with this sentiment. According to their work, by removing opportunities to find work through the internet, consenting sex workers can not safely advertise without the need for a pimp, screen potential clients, and use more reliable electronic payment methods under FOSTA/SESTA laws. This also impeded on a sex workers right to free speech on the internet, further throwing the internet into censorship, and silencing the voices of a community that has already faced so much adversity. This puts into perspective the possible unintended side effects that can occur from well meaning work.

It is especially important in this type of research to think carefully about how a disruption may effect unintended targets in addition to traffickers. Efforts to combat sex trafficking may push the enterprise further underground, place persons being trafficked in a violent situation, and/or cause traffickers to develop new and better ways to avoid detection and recruit more victims, diminishing the number of opportunities for intervention. It is also important to respect the victim's agency. Persons experiencing trafficking should be empowered and motivated to leave the trafficking network [46]. Also, once a trafficking survivor is removed from the network, they will still be at risk for being re-trafficked. Models, therefore, need to be constructed in a way that understands the impact of their decisions on the victims. In order to accomplish this, quantitative researchers must effectively collaborate with qualitative researchers and domain experts. For example, Sharkey et al. [47] discuss a framework to accomplish this collaboration. We have applied such a collaboration in the sense that the codebook, which the qualitative researchers used to create networks, was produced through knowledge both within the trafficking domain and in the operations research domain. The NLP methods were implemented to mimic the use of the codebook in constructing the network data.

Chapter 3

Research Design and Methods

Our framework refers to a series of machine learning and natural language processing techniques which transform the data several times and produce the results of our experiment. Federally prosecuted case files, dockets, and indictments, were used to train and test the model and the created methods. Rule-based pattern matching was also used as a sort of filtering process to produce the output. This chapter provides an explanation of the framework in 5 parts: 1 PDF to text conversion, 2 Text cleaning, 3 Model training, 4 Node extraction, and 5 Relation extraction. Steps 2 - 5, although described with examples from the data available, may be more broadly applied and are not domain specific.

3.0.1 Goals

Through this work we hope to provide a framework that performs some steps of qualitative coding automatically and outputs node and relation spreadsheets to produce computer-generated networks. Human assistance will still be required, but at a much more "hands-off" level. Ultimately, we would like to automatically generate network models from the information acquired from the automated coded spreadsheets. In our work, we aim to analyze the resulting outputs for their overall utility, as well as perform centrality measurements (see Chapter 4 for details). In this work, we focus on victims, traffickers, and hotels in the text. This emphasis allows us to map out how people move through the network to different locations while also capturing interactions. We focus on these three things as our preliminary functionality because it describes the physical world and relationships. This is our first priority of information since it constructs the networks.

3.1 PDF To Text Conversion

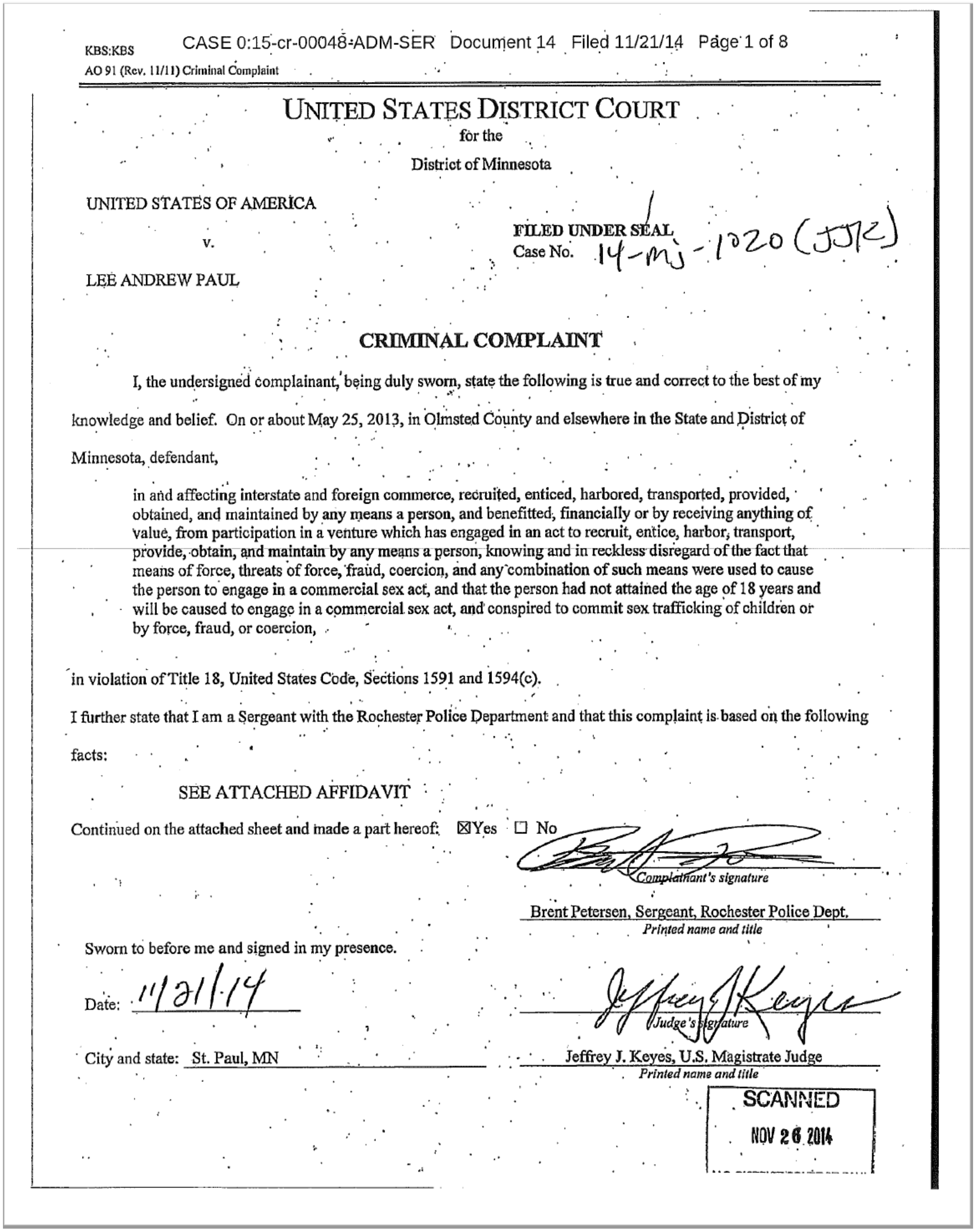
The original format of the prosecuted case files, dockets, and indictments came to us as scanned PDFs of raw documents. These documents included various degradations such as lines, various type face, handwriting, black speckles, check boxes, stamps, and various areas of writing on the page. To make the files easier to work with, as PDFs are a proprietary format, the file PDFs were converted to images using Image Magick version 7.0.

This was also performed to transform the files into the proper input format for Tesseract OCR, an optical character recognition software that translates images' text to actual computer text. The PDF images are then assigned to their own folder as a list of ordered image documents. Figure 3.1 illustrates the general condition of the scanned case files. Note the many imperfections throughout each document as well as varying text positions and fonts.

Next, the new file images are ready for redacted mentions to be placed back into their original location on the page. This is necessary due to the presence of white-out on the original case file documents. Areas on the case files are whited out due to a desire to withhold a name from whomever accesses the document. This, however, results in several large "whited-out" gaps on the page.

Failing to properly insert the word "redacted" into the empty parts of the image where the whited-out portions are located causes great errors in translation following the use of the image to text set up. To perform this step requires the assistance of a human individual. A screen pops up with the front page of the case file image. The user then left clicks the areas that have been whited-out and the text "redacted" is placed in the clicked area. The user may press the z button to undo an insert if they have made an error and move the pages with the left and right buttons. The user moves through each page, performing the same steps until all the pages have had redacted inserted into any intentionally whited-out spots. Figure 3.2 provides an example of what the insert redacted step looks like for a human performing the task.

Tesseract OCR version 5 is then called upon to convert the images of the scanned documents into text. Please refer to Table 3.1 to execute the steps which convert the PDF to an image and the image to text.



KBS:KBS CASE 0:15-cr-00048-ADM-SER Document 14 Filed 11/21/14 Page 1 of 8

AO 91 (Rev. 11/11) Criminal Complaint

UNITED STATES DISTRICT COURT
for the
District of Minnesota

UNITED STATES OF AMERICA

v.

LEE ANDREW PAUL

FILED UNDER SEAL
Case No. 14-mj-1020 (JJK)

CRIMINAL COMPLAINT

I, the undersigned complainant, being duly sworn, state the following is true and correct to the best of my knowledge and belief. On or about May 25, 2013, in Olmsted County and elsewhere in the State and District of Minnesota, defendant,

in and affecting interstate and foreign commerce, recruited, enticed, harbored, transported, provided, obtained, and maintained by any means a person, and benefitted, financially or by receiving anything of value, from participation in a venture which has engaged in an act to recruit, entice, harbor, transport, provide, obtain, and maintain by any means a person, knowing and in reckless disregard of the fact that means of force, threats of force, fraud, coercion, and any combination of such means were used to cause the person to engage in a commercial sex act, and that the person had not attained the age of 18 years and will be caused to engage in a commercial sex act, and conspired to commit sex trafficking of children or by force, fraud, or coercion,

in violation of Title 18, United States Code, Sections 1591 and 1594(c).

I further state that I am a Sergeant with the Rochester Police Department and that this complaint is based on the following facts:

SEE ATTACHED AFFIDAVIT

Continued on the attached sheet and made a part hereof: ☒Yes ☐ No

Complainant's signature

Brent Petersen, Sergeant, Rochester Police Dept.
Printed name and title

Sworn to before me and signed in my presence.

Date: 11/21/14

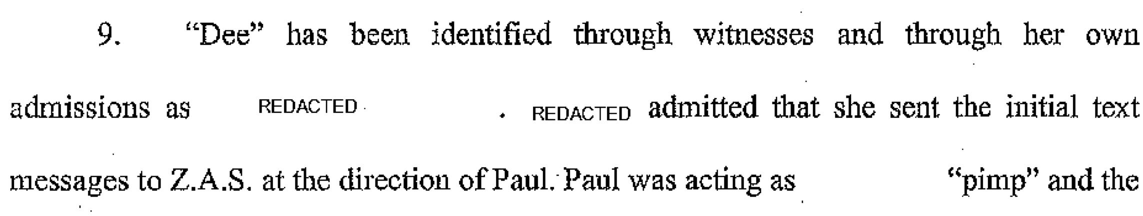
Judge's signature

City and state: St. Paul, MN

Jeffrey J. Keyes, U.S. Magistrate Judge
Printed name and title

SCANNED
NOV 26 2014

Figure 3.1: The case file data prior to pre-processing steps. Note the many details on the page.



9. "Dee" has been identified through witnesses and through her own admissions as REDACTED . REDACTED admitted that she sent the initial text messages to Z.A.S. at the direction of Paul. Paul was acting as "pimp" and the

Figure 3.2: Words are redacted to protect the identity of individuals within the document. This, however, causes errors and difficulty during text analysis as this causes the subject to be missing from the sentences.

	Description
1	Convert PDF scanned image to image
2	Extract text from the image with Tesseract OCR
	Command
1	`convert -density 150 {inPath} {outDir}/temp.png`
2	`tesseract.exe --dpi 150 {inPath} {outPath}`

Table 3.1: Command line arguments to perform conversions.

Each text file is then joined together as one solid body of text relating to their own case file, respectively. After the text conversion, the text needs to undergo object coreference resolution. Object coreference resolution is an NLP tool that can replace all pronouns with proper nouns [48]. This is an important cleaning step that demystifies the account of what occurred in the case file and to or by whom. For example, now instead of capturing "his phone", the extraction functions can pick up "proper name's phone". Then, when the phone node is extracted, the name of the person who owns the phone can be part of the information, just as it is in the human-coded spreadsheets. Since computers can not keep track of context clues that indicate who the pronouns are referring to, object coreferrencing resolution offers an efficient solution to keep track of who does what and who owns what. The module that performs this in Python, Neuralcoref [49], need only be added to the pipeline once for use and called once using the desired coreferrencing text as the argument. Afterward, the loaded text document will be coreferrence resolved. Following this step, the metadata from the body of text is removed as well as non-ascii characters. The resulting cleaned text is saved as its own text file for use in the training module.

3.1.1 Setting Up the Training Data

The data will not be immediately ready for model training and must be further cleaned and formatted before model training. The first step in this process is to define the labels. Recall that for the purpose of this work, we will only examine the victim, trafficker, and hotel labels, however, the vehicle, phone, and cyber labels are also included for future work. All training words are contained in their own json file and include the words that indicate a label should be used. For example, training words for the victim label contain "Z.A.S.", "J.E.", "victim", and all permutations of two and three letter abbreviations as all victim's identities are kept private through the use of abbreviations. A function was created to get each new token entity based on if the word is found in the training words. If it is, it is assigned the appropriate label and returned. Entity spans were also trimmed from the set, which means that white space is removed from the beginning and end of each training word. The final two functions run the load document and get training data commands.

3.2 Model Training

The model is trained to recognize specific categories of words. The words are victim names, trafficker names, phones, supplies, vehicles, and anything else in the text that is labeled in the human-coded node extraction spreadsheets. The words are assigned the same label they have in the human-coded spreadsheet. The training words were saved as a separate json file. In this work, the labels we have used are "HOTEL", "TRAFFICKER", and "VICTIM". This means that these categories of words are extracted from the text and assigned the label they belong to whenever they are mentioned in the text. For example, a mention of "proper hotel name" in a sentence is extracted as "proper hotel name: HOTEL" where "proper noun" is the node name and "HOTEL" is the label assigned to it. The sentences the words were extracted from are saved in a data structure to be used in the training step.

The labels were added with the help of spaCy, a natural language processing tool. We used the EntityRecognizer function to perform the labeling task. The EntityRecognizer comes with a default entity recognizer, but can be added upon through additional model training [29]. The labels used in the human-coded spreadsheets were assigned as additional entity types to the existing SpaCy NER (Named Entity Recognition) model with the intention of using this model to label node words in text files. We set up the pipeline and entity recognizer to train the model to recognize new entities

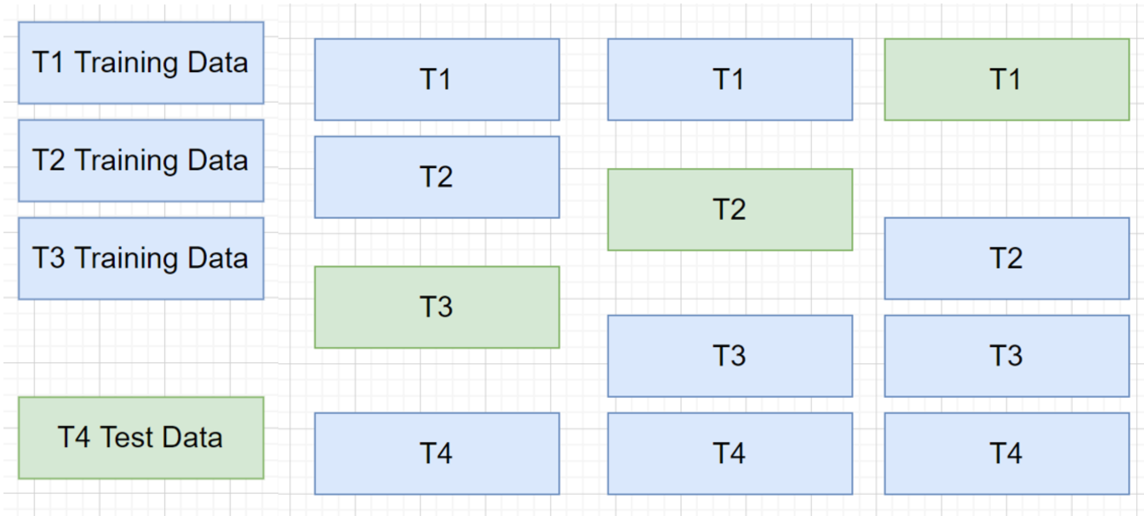


Figure 3.3: Four iterations of LOOV. T4, T3, T2, and T1

by feeding it labeled keywords from the human-coded excel spreadsheets. Other NLP pipes were turned off during the training so that only the NER pipe would be affected and added to.

3.2.1 Validation

After the model training process was finished, we performed validation testing. Leave-one-out cross-validation was the chosen method as the data set was markedly small. Leave-one-out cross-validation rotates each data point as the test data and the rest of the data set as the training set until all data points have been used as an iteration of the test data [50]. Refer to Figure 3.3 for a visual representation of leave-one-out cross-validation. Notice the blue training data in each of the four iterations, and the green testing data. In our experiment, each case file or indictment/docket was used as the test data at least once (green block) and used as the testing data (blue blocks) three times. In addition to this, we also performed leave-one-out cross-validation on 9 other document sources which included indictments and dockets. The case files are fully structured complete accounts of a criminal event detailing a victim and trafficker. The nine dockets and indictments were combined as a single document in the text cleaning process. The information in these documents are not as whole as the case files.

Node Name	Node Type
trafficker's name	TRAFFICKER

Figure 3.4: A sample entry of a node and label pair on a spreadsheet.

	A	B	C
1	**Arc Start Node**	**Arc End Node**	**Arc Description**
2	trafficker's name	trafficker name's phone	Admitted to sending recruitment messages

Figure 3.5: A sample entry for a relation extraction arc. Entries are directional from Start Node to End Node. Start and End Nodes are not bidirectional. An entry is classified as incorrect if what is supposed to be a start entry is placed in the end column and what is supposed to be an end entry is placed in the start column.

3.2.2 Expected Output

There are two excel spreadsheets that are produced in the output. The first spreadsheet is known as the node data extraction sheet. It contains one column for the node name and another column to the right of it that contains the label. The node name column possesses all of the proper names of the nodes extracted from text. The column to the right of this shows the label that the proper node name belongs to. In essence, this sheet is like a categorization of the nodes. An example entry appears in Figure 3.4.

The second excel spreadsheet is the relationship arc data spreadsheet. It contains three columns. The first column is called the arc start node and the column to the right of it is called the arc end node. The third column to the far right of these is called the arc description. The arc start node column contains the proper names of nodes that start an interaction with the proper noun names in the arc end node column. The arc description column on the far right minimally describes the interaction. For instance, if in a case file, a trafficker uses their phone to call someone, you might have an entry that looks like Figure 3.5. Finally, a computer-generated network is created from the relation extraction spreadsheet.

3.3 Node Extraction

3.3.1 Get Proper Name

One challenge we faced in extraction, is how to retrieve proper names of items in the case file text. We do not want to simply extract the word "phone", we want to extract that word and whom it is owned by. Since the model is trained on words only, descriptions are not extracted nor are they always in a proper format for extraction. This was solved through scaling the dependency trees (See Figure 3.6 for an example of a dependency tree.) of the labeled word and looking for dependencies. If the target word is a **"VERB", we save the verb word in the match string. Then we look at that node's left children. The word immediately to the left is appended as another part in the match string. If this child has children with a "NOUN" part of speech or that child's dependency is a "dobj" or "compound", then that word is integrated into the target word as well.**. This process is shown in the psuedocode included below:

Algorithm 1 NOUN-VERB-NOUN Pattern Capture

```
procedure getMatchString(nlp, verbString, matchString)
  for t ∈ s do
    if t.text = verbstring then
      for a ∈ t.children do
        if a.position = VERB then
          a.text ← a.verb
          aLefts ∈ [t.text ← t ∈ a.lefts]
          if len(a.lefts) > 0 then
            a.part ← aLefts[0]
            matchString ← matchString + aPart + aVerb
          else
            matchString ← matchString + aVerb
            for b ∈ a.children do
              if b.pos = NOUN&(b.dep = dobj||b.dep = compound) then
                bVerb ← b.text
                bLefts ← [t.textfortinb.lefts]
                if len(bLefts) >= 1 then
                  bPart ← bLefts[0]
                  matchString ← matchString + bPart + bVerb
      matchString ← locationwhere + matchString
  return matchString
=0
```

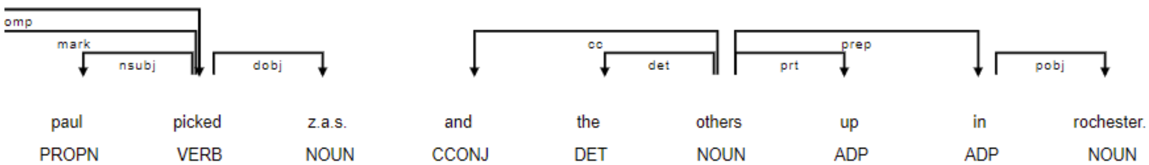


Figure 3.6: A sample dependency tree that is broken down in the figure below.

Text	***Parts of Speech***	***Dependency Abbr.***	***Dependency***
paul	**PROPN**	**nsubj**	**noun subject**
picked	**VERB**	**xcomp**	**open predicate complement**
z.a.s.	**NOUN**	**dobj**	**dependent object**
and	**CCONJ**	**cc**	**consonant**
the	**DET**	**det**	**determiner**
others	**NOUN**	**none**	**none**
up	**ADP**	**prt**	**particle**
in	**ADP**	**prep**	**preposition**
rochester	**NOUN**	**pobj**	**possessive object**

Table 3.2: An example of a sentence visualized as a dependency tree (Figure 3.6) and its parts of speech labels (Figure 3.2).

3.3.2 Proper Names List: Clusters, sets, replacements

To handle the problem with typos of victim/trafficker target words, a similarity scoring function was employed. If victim/trafficker words were similar to each other via a similarity scoring system, they were saved into a data structure and clustered together. The victim/trafficker word spelling with the highest frequency in each cluster was taken as the correct spelling and used to replace misspellings in the node list.

3.3.3 Unique Nodes and Export

Another problem that arose was that the code did not always extract proper names of hotels. Instead, it would extract words such as "the hotel" or "the motel". Recall that the lack of formality causes problems during node extraction. We only want to extract a hotel node once and we want its proper name to be included in the extraction. A function was created to handle this issue. The function used a series of cases to extract the hotel's proper name through a short series of grammar rules. It is likely that there are edge cases for this function, however, during our experiments we did not find any. Finally, the lists are pared down to unique mentions only and exported as a csv file.

3.4 Relation Extraction

3.4.1 Get Original PDF Text

The first step in relation extraction is to read in the data needed for transformation. The sentences exported from the set up training data step are imported. The text from the original PDF file is read in as it will be used to locate sentences within the PDF for highlighting.

3.4.2 Create Victim Trafficker List

In this step, the correct spelling for the trafficker and victim node are extracted and the relation is appended to each. Because the trafficker and victim relation is always the same, the description "trafficker & victim" is used for each trafficker and victim pairing. This function also ensures that there aren't multiple typoed spellings for each entity.

3.4.3 The Highlight Folder

A folder is created to hold all of the highlighted mentions from the sentences as they are found in the original PDFs. A highlighted mention is the exact sentence in the original PDF where the node instance was taken from. The output provides a link to the location on the PDF where the source sentence is located. For ease of visibility, the source sentence is highlighted on the PDF page. **This enables the human user to have an easy way to identify which sentence where the description was found is.** This in turn allows the human user to clear up any ambiguity from the description by reading what the original sentence says, increasing this experiment's utility. The function also retrieves the page number where the text was found to further assist the human user.

3.4.4 Hotel Relation and Name Extraction

A pattern matcher is used to extract the **proper noun, verb, proper noun** relation from text. This pattern is found when some entity is acting upon another in the text. From here, the hotel relation and name extraction function is used. It starts by instantiating the pattern matcher, and then loops through each node, sentence, and PDF match in the test array, but only for sentences with hotel words in them. For each match found through the pattern matcher, a match string is created. This gets the proper name of the entity found. Next, the functions that get the page number and the highlight link are ran. The function returns the two entities, a hotel name, a match string, a page number, and the highlighted pdf path.

3.5 Summary of Methods

The methods began with PDF to text conversion. From here, the data is cleaned and made ready for model training. After model training is performed, the new model is used to extract nodes from the text. Following this, arcs are extracted between nodes. Steps must be performed sequentially as each process is dependent on the last. This method is just one of many that can be attempted. Note that we are using a knowledge-based technique called "rule-based matching" to extract arcs and a trained model to extract nodes. Refer back to Figure 1.4 for a visual description of the framework complete with labels pertaining to sections discussed in this chapter.

Chapter 4

Analysis and Results

4.1 Introduction

The chapter is divided into two segments: Analysis and results. In the analysis section, we will discuss definitions of important terms, our qualitative scoring system, and our scoring criteria. In the results section, we cover our experimental results including, node extraction, relation extraction, and network comparison. We will go over our use of traditional and non-traditional approaches to analyze the results. Our tests are performed on two types of data and our model is trained in various permutations of the data sets to test for transferrability, effectiveness, and ideal data conditions. Finally, we will discuss potential improvements.

4.2 Definitions and Formulas

In this work, a **correctly predicted output**, is any output that matches our ground truth under a specific criteria. An **incorrectly predicted output** is any output that does not meet the criteria stipulated. Incorrect outputs are decided according to a criteria detailed later on in this chapter. The total number of entries in a human-coded spreadsheet is referred to as the **total to extract**. Accuracy is measured as the total number of correctly predicted outputs over the sum of incorrect predictions and the total number of extractions to get from the text to match the ground truth. When there are no incorrect predictions, the accuracy is the same score as the precision. The resulting node and relation extractions have a column name of the letter “T” for case files or “D” for dockets and indictments, and then an ordered number. This stands for a distinct case file or docket indictment pair which consists of unique data specific to that case. We define accuracy of our methods to be:

$$Accuracy = \frac{CorrectlyPredicted}{IncorrectlyPredicted + TotalToExtract}$$

4.2.1 Nodes, Arcs, and Actors

A node is defined as any entity in the text that bears influence in the events being described. These typically include people and items that the people use. An arc is used to indicate a relationship between two nodes. The relationship can be as simple as a person and their phone, or as complicated as a victim and trafficker relationship. The important information is that we define an arc as two nodes related to each other in some way. Finally, an actor refers only to the people within the text.

4.2.2 Case Files

The case files are a collection of 4 different complaint documents that are defined as case files. Case files are complaints that detail the criminal events of a sex trafficker (or traffickers). As stated before, the initial condition the case files were given was scanned and filled out raw PDFs. The complaints detail the criminal events a sex trafficker or sex traffickers have committed over a period of time. The case files do contain graphic content and our research group has used the utmost discretion in handling the material. All personnel assigned to work with these documents have undergone appropriate BRB training and have been given access to counseling resources. The case file documents range in length from 8 to 17 pages and were filed before the U.S. District Court of Minnesota over the period of 2009 to 2016 [51]. All sex traffickers mentioned were prosecuted. The locations of the criminal events include cities in the state of Minnesota and the bordering states.

4.2.3 Indictments and Dockets

We also had additional access to other potential sex trafficking networks but the source of each of these were not based on case files. In particular, the source included indictments, superseding indictments, and dockets. These documents also came to us as scanned PDFs, although the document condition was objectively better. The pages included separating lines, dates all throughout, numbered enforcement codes, and writing within several organized boxes. The material covered within them is far more concerned with the legal side of things and not so much a line by line account of the criminal event. A docket is a short summary that details all proceedings, filings, and deadlines in a case [52]. An indictment is a formal accusation with basic information about the criminal offense they are believed to have committed. They consist of one page or more and contain fewer graphic details than the case files. A superseding indictment has additional charges added to the original indictment and replaces it [52]. Indictments and dockets do not include complaints. These documents were from the same period of 2009 to 2016 as the case files and the locations of the crimes were also in Minnesota and neighboring states. The indictments and dockets are also distinct documents filed before the U.S District Court of Minnesota, just like the case files were.

4.3 Criteria for Correct and Incorrect Outputs

One unique type of criteria test was used to classify the node and arc outcomes as correctly predicted outputs or incorrectly predicted outputs. Our ground truth are the human-coded spreadsheets of each case file. The unique criteria is known as generic output criteria. This analysis has simple and lenient rules for scoring correctly predicted outputs. A correctly predicted output for the generic output criteria must extract nodes and arcs that are correct to the account of the story in the case file or must be important to the overall meaning of the text, whether or not the correctly predicted output is contained in the human-coded spreadsheets in the exact same manner. This means that correctly predicted outputs do not need to be exact matches to the ground truth to be a correct assessment. Use of nicknames are okay to extract and include in the node output but by the generic output criteria's standards, are still counted as an incorrect answer. In node extraction, an incorrectly predicted output can happen when an incorrect label is assigned to a node or if the extracted node is of no importance to the case. An incorrectly predicted output can be an unimportant node or relation extraction. In relation extraction, a predicted output can be incorrect if the relation extracted does not lead a human user to a logical outcome of the event or if the results are gibberish or nonsensical. An example of a relation being incorrect may have a correct arc but the text connecting the two nodes may not truly indicate a relationship that should lead to an arc being present. The generic output criteria emphasizes that we care about the resulting network, allowing for more information and detail under reasonable flexibility.

In node extraction, we do not need the extracted node to be the exact *same* one in our ground truth with the exact *same* label. This outcome could not be feasible. Node and relation extraction are not an exact science, and the generic outcome criteria accounts for that notion by not expecting exact matches. In this work, we recognize the limitations of our methods and the role of human coders in analyzing very specific aspects of sex trafficking networks.

Although our analysis technique is task specific and unconventional, others have used similar equations and criteria to determine correctly outputted data [53] and to evaluate qualitative classifications with human assisted methods [21]. Because our input and output ratio is not one to one, we cannot use traditional binary classification scoring, therefore, we have proposed the generic output criteria as a substitute. As touched on previously, when a node that should be in the ground truth is found, the number for total to extract from the ground truth is increased to reflect the newly identified node as a correction to the ground truth. For example, we may see an additional hotel node in the computer-coded node spread sheet that should be in the ground truth but was missed by a human-coder. The total to extract from the ground truth increases by one to reflect the correction. View the updated accuracy equation below, where "n" is the total number of extra nodes or arcs found by the computer that were deemed that they should have been identified.

$$Accuracy = \frac{CorrectlyPredicted}{IncorrectlyPredicted + TotalToExtract + n}$$

4.3.1 Generating Network Graphs for Human-Coded Comparison

A network graph is a visual representation of nodes and arcs. The nodes are situated on the page and the arcs stem from a node to another node that it has a relation with. The arcs in our networks are directed, with a clear and specific start and end node. The node that initiates an interaction with another node is the start node, while the node being acted upon is the end node. The human-coded networks are drawn and placed next to the computer-generated coded networks for comparison.

4.4 Degree-Centrality Measurement Discussion

Degree-centrality is measured to approximate which node is on the shortest path between all other nodes. This tells us which of them can deliver flow to the most nodes [54]. When comparing the human-coded network and computer-coded network, if there is a node with the same degree centrality, we may understand that the computer code is finding the same important nodes. If two networks have the same pattern of degree centrality, then this can reveal more information about trafficking networks. Additionally, the networks can show us if a node has a larger than average number of connections, so that we can look closer and draw conclusions on that node. It is performed for each of the 4 case file documents that have undergone leave-one-out cross-validation. In this work, the name of the node with the highest betweeness-centrality will be listed in the results section.

4.4.1 What Good Scores Typically Look Like for This Domain

A good accuracy scoring in this field varies greatly under several different conditions and performance. We can also measure performance in terms of F1 scores, which is a combination of precision and recall. A F1 score is defined as the harmonic mean of precision and recall where recall is the amount of relevant results found divided by the total amount of existing relevant results and precision is the amount of relevant results found divided by the total amount of results retrieved [55]. The F1 score is a more intuitive mean for this problem than the arithmetic mean because it is a ratio of precision and recall. A F1 score's range is between 0 and 1 with 0 being the lowest score, and 1 being the highest. The following three equations characterize precision, recall, and the F1 scores.

$$Recall = \frac{relevantPredictions}{TotalToExtract}$$

$$Precision = \frac{relevantPredictions}{TotalNumberOfExtractedItems}$$

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

Scores are dependent on the work's domain, parameters, availability of data, quantity of data, and complexity, among other criteria. Even in Wang et. al. [56], F1 scores varied between 0.3423 to 0.5530 among several relation extraction techniques including rule-based approach, statistical approach with and without filters, and filters only, and their results were state-of-the-art for their time. In Liu et al. [57], researchers used NovelTagging, OneDecoder, MultiDecoder, and GraphRel with both 1 and 2 parameters and achieved a F1 score of 0.619 as their highest score. Good F1

generic output criteria: Nodes	***T1***	***T2***	***T3***	***T4***
correctPredictions	**7**	**4**	**5**	**4**
totalToExtract	**7**	**4**	**5**	**5**
totalOfExtractedItems	**7**	**5**	**8**	**4**
precision	**1**	**0.8**	**0.625**	**1**
recall	**1**	**1**	**1**	**0.8**
F1	**1**	**0.88**	**0.769**	**0.88**
accuracy	**100%**	**80%**	**62.5%**	**80%**

Table 4.1: Generic Output Criteria Node Results

scores are typically above 0.50, however, because our outputs are not one to one, our method of defining correctly predicted and incorrectly predicted outcomes has some subjectivity to it. Our own node extraction task for our four case files acheived high F1 scores while our relation extraction task performed sub par. Relation extraction is always improving, but, it is still difficult to increase the extraction scores much.

4.5 Leave-One-Out Cross-Validation

4.5.1 Generic Output Criteria Results

Under the generic output criteria, the node extraction task accuracy pictured in Table 4.1 scored above the modern baseline ranging from 62.5% to 100% with a group average of 80%. Relation extraction pictured in Table 4.2 performed poorly with results ranging from 16% to 44.4% with an average of just 31%.

4.5.2 Leave-One-Out Cross-Validation for Indictments and Dockets

The first node extraction entry D1 is not included in the analysis as this did not contain a proper node output. Node analysis was the only test performed on the indictment and docket data because the

generic output criteria: Arcs	*T1*	*T2*	*T3*	*T4*
correctPredictions	**5**	**1**	**2**	**2**
totalToExtract	**12**	**5**	**6**	**7**
totalOfExtractedItems	**7**	**2**	**3**	**2**
precision	**0.71**	**0.5**	**0.67**	**1**
recall	**0.42**	**0.2**	**0.33**	**0.29**
F1	**0.52**	**0.285**	**0.44**	**0.44**
accuracy	**35.7%**	**16%**	**29%**	**44.4%**

Table 4.2: Generic Output Criteria Arc Results

generic output criteria: Nodes	*D1*	*D2*	*D3*	*D4*	*D5*	*D6*	*D7*	*D8*	*D9*
correctPredictions	**-**	**1**	**2**	**2**	**3**	**0**	**0**	**1**	**3**
totalToExtract	**-**	**1**	**3**	**3**	**3**	**1**	**2**	**1**	**3**
totalOfExtractedItems	**-**	**4**	**4**	**3**	**5**	**2**	**4**	**2**	**5**
precision	**-**	**1**	**0.66**	**0.66**	**1**	**0**	**0**	**1**	**1**
recall	**-**	**0.25**	**0.5**	**0.66**	**0.6**	**0**	**0**	**0.5**	**0.6**
f1	**-**	**0.4**	**0.569**	**0.66**	**0.75**	**0**	**0**	**0.66**	**0.75**
accuracy	**-**	**25%**	**40%**	**50%**	**60%**	**0%**	**0%**	**50%**	**60%**

Table 4.3: Indictments and Dockets: Generic Output Criteria Node Results

results had some viability to them although there are nine entries (Shown in Table 4.3). Fortunately, most of the node spreadsheets were in good condition so that we can demonstrate scores on a different type of data. The generic output criteria had variable scores ranging from 0% to 60% and a total average of 35%.

4.6 Comparison of Centrality of 4 Case Files

Results are mixed as far as the comparison of highest centrality nodes between human and computer-generated networks. Three out of the four tests had computer-generated networks that contained all of the highest centrality nodes of the human-generated results but did not produce exact results. The biggest problems stem from the computer-generated network's lack of hotel nodes.

Network N2 (Figure 4.2) is missing the two hotels from the ground-truth such that the computer-

Highest Centrality Nodes	***N1***	***N2***	***N3***	***N4***
Human-Coded	**“Minor”**	**“McHenry”, “J.E.”, “N.A.”**	**“G.S.”**	**“Paul”, “Z.A.S.”, “K.A.J.”**
Computer-Coded	**“mckie”, “Victim”**	**“McHenry”, “J.E.”, “N.A.”**	**“Geddes”, “Victim”**	**“K.A.J.”, “Z.A.S.”**

Table 4.4: Centrality Measurement of Four Case File Networks. Each network contains the node(s) with the highest centrality.

coded network is not complete. Another noticeable difference in network N2, is that the trafficker node is called “Gilmore” in the human-coded network and “mckie” in the computer-coded network although one refers to a trafficker and one refers to a “bottom”. Because the bottom shows characteristics of the trafficker, distinctions between a trafficker and bottom is a nuance that is not built into the model. Additionally, distinctions between actor’s nicknames or surnames and the actual name in the ground truth are not built into the model either. These are the types of tasks that have only been performed by humans.

Network N3 is interesting because the computer-coded network replicates the top half of the human-coded data, although it fails to capture the hotel node pictured in the human-coded network. The same scenario is seen in N4 (Figure 4.4) with the missing hotel nodes.

In N5 (Figure 4.5) the victim nodes of the computer-coded network does not have the “Paul” node connected to any of the hotel nodes. This causes the network to only capture the two victim nodes as having the highest centrality when it should be all three, like in the human-coded network.

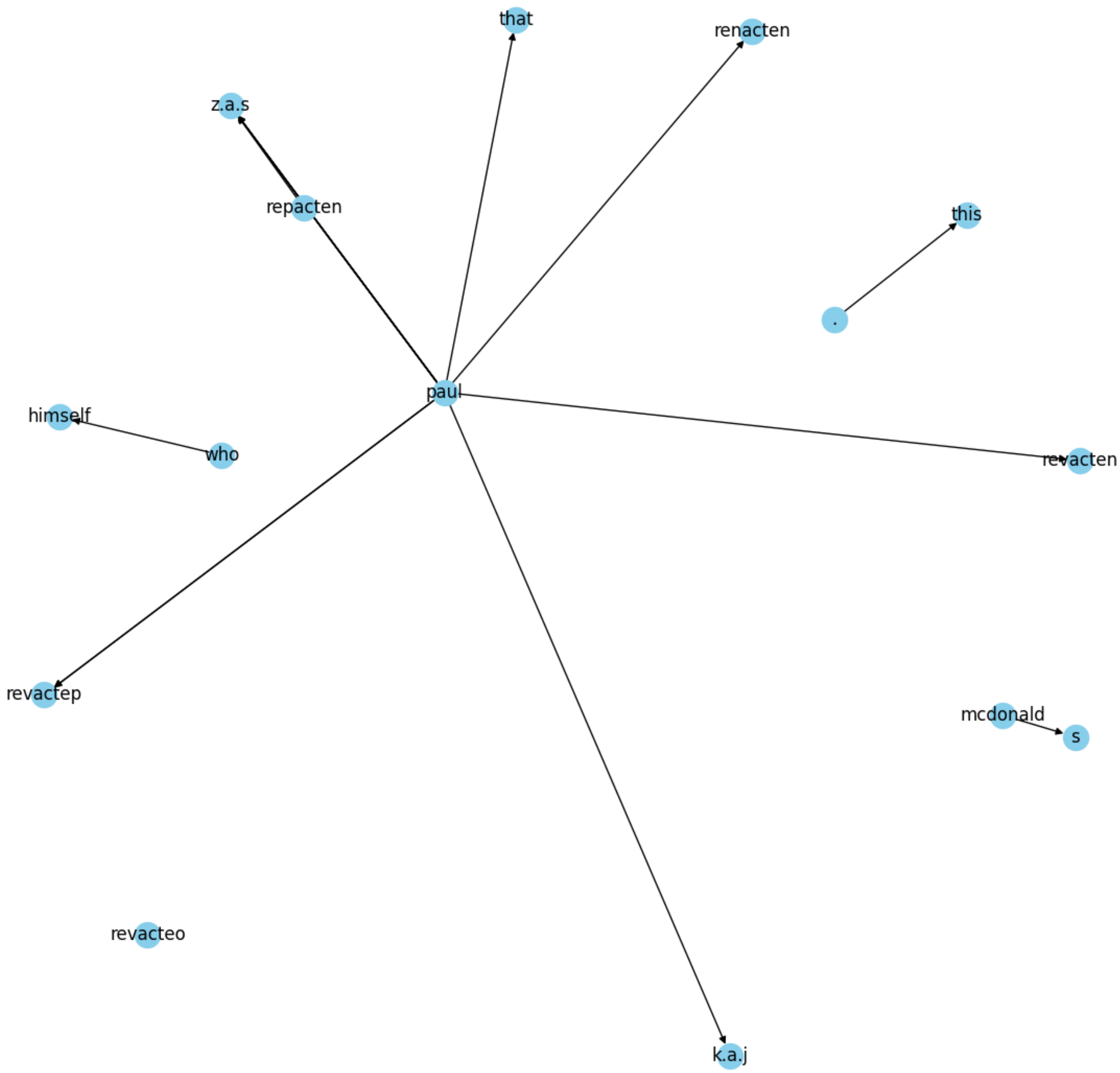


Figure 4.1: N1: A computer-generated network that was automatically produced in the output. Note the erroneous entries in the disconnected graph. The node with the highest centrality is pictured in the center.

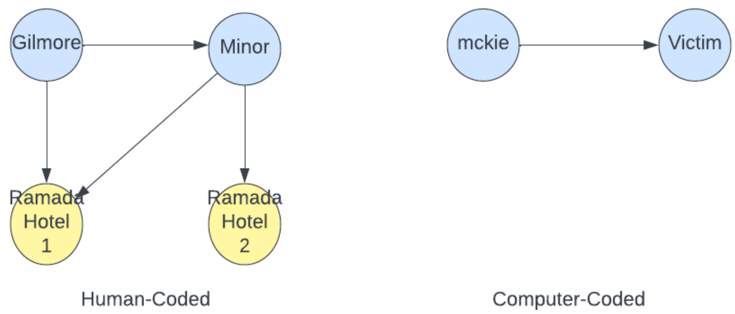


Figure 4.2: N2: The computer-generated network is missing the hotel nodes found in the human-coded network.

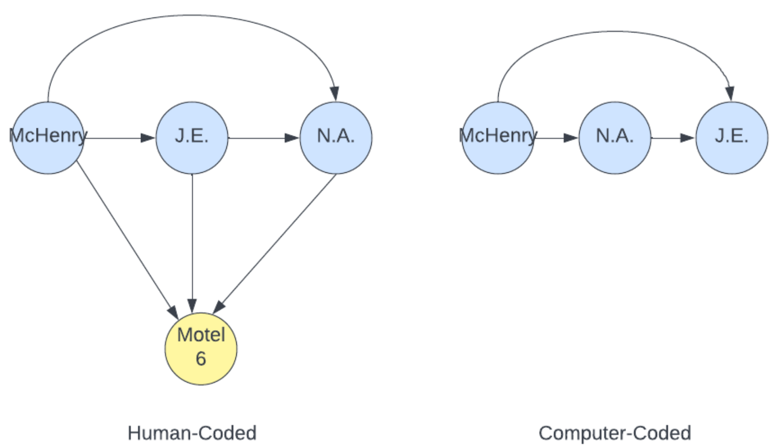


Figure 4.3: N3: The computer-coded network's nodes are all of the same degree centrality. The computer-coded network captures this but misses the hotel node found in the human-coded network.

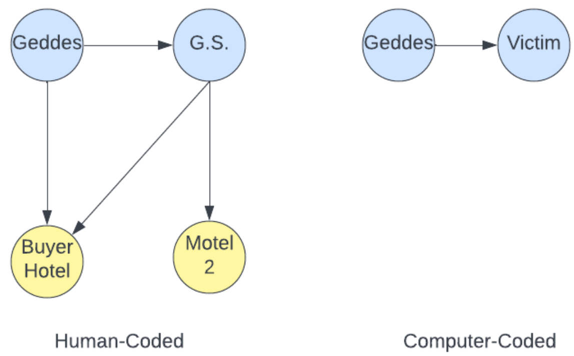


Figure 4.4: N4: The victim, G.S., has the highest degree-centrality in the human-coded network but is lacking in the computer-coded one.

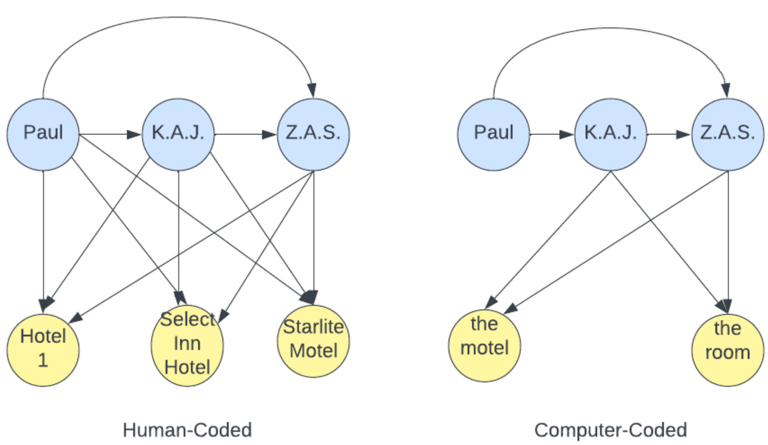


Figure 4.5: N5: In this network, the trafficker and victim nodes all have the highest degree-centrality. The computer-generated network did not capture this information and instead missed the "Paul" node as part of it's calculation for highest degree-centrality.

4.7 Train on Case Files, Test on Other Sources/Train on Other Sources, Test on Case Files

To test the transferability of the node and relationship arc tasks, we performed a sort of cross training and testing, that is, in one experiment case files were used as the training set and tested on the indictments and dockets. Likewise, we also performed an experiment where we trained the model on indictments and dockets, and tested on case files. The results of this test were non-viable. Outputs included too many duplicates, exceedingly short spreadsheets, gibberish, and in some cases, no output at all. There are several reasons this may be occurring. The primary reason is that dockets, superseding dockets, indictments, and complaints are all highly structured data that are composed in their own way, respectively. The structure of a complaint is not structured like a docket, and a docket is not structured at all like a complaint. Although structured data is easier to work with, the trained models are not interchangeable with data structured differently than what it was trained on, therefore, the training process should contain data of different types to gain better results.

4.8 Conclusion of Results

Node extraction performed far greater than relation extraction under generic output criteria. Node extraction likely performed better because it used a pre-trained NER model with a pipe containing our labels on it. The machine learning method was more reliable and produced more desired results. With relation extraction, we used a knowledge-based pattern matching technique. Although the data was highly structured, as it would need to be for pattern matching technique, the results were still unimpressive. In addition to this, the framework is not transferable to other documents. Future work will look at usage of new relation extraction repositories in addition to a training process that includes all court documents. Network structures were different among computer and human-generated networks. Although, computer-coded networks did mostly contain the same highest degree nodes as the human-generated networks, results were not exact.

Chapter 5

Discussion and Conclusion

5.1 Limitations

This kind of research is not without its difficulties and there are ways to remedy the current process. Additionally, a totally new process can be tried on the data. One of the difficulties in working with this type of text data was that there was not a one to one ratio between inputs and outputs. This made scoring the results unclear at times and injected bias into our process. The generic output criteria was meant to reduce bias as much as possible, however, more traditional applications are able to use computer tools to determine solutions without having to use human inspection and similarity measures to identify correct or incorrect predictions. Additionally, this work only examined three labels and relation extraction between victim, trafficker, and hotel location.

5.1.1 Limited Data

One reason for the resulting node extraction scores is that there just was not enough data to train a NER model effectively. Without enough data, the model didn't have enough examples to learn from. To put this in other words, the model needs enough instances of the training words used in different ways to decide with enough confidence that the word in the text is a "trafficker" or "hotel" because the model had enough instances of this word being used in a variety of contexts. Essentially, the model performs better when it has seen a label word used in a specific context during the training process.

5.1.2 Transferability

An important reason why our relation extraction performed below standard was because of the lack of transferability between file types. Recall that the data used in our work consisted of case files, dockets, and indictments. All three of these data types are structured differently, that is, the way information is situated on the document itself varies greatly across each type of data. The rule-based matching technique for relation extraction was created for case files, as they contain the most information about a criminal event. These rules, however, did not hold up when applied to an alternative data format, namely indictments and dockets. A series of rules for different types of documents would need to be added to the current rules to see an improvement in relation extraction scores.

5.1.3 Additional Collaborations

The node and especially the relation extraction tasks did not perform up to par because there was a disconnect between traditional analytical techniques and sociology and social networking strategies. Because our data consisted of interpersonal relationships, it would make sense to consult with those who normally use their expertise to perform social network analysis. This additional information could bridge the gap between an out-of-the-box technical method and an effective process that is more suited for the data and tasks used in this work.

5.2 Areas of Improvement

There is a question of whether or not this type of NLP is the right tool for *this* job, however, our experience is only one way to do qualitative coding on case files, dockets, indictments, and even other court related data. The domain remains open for improvement and continues to be rife with opportunity for growth. This method could have performed better had there been an adequate amount of data to work with. The node extraction step would benefit from the increased number of examples used in the training set and the relation extraction step could integrate machine learning into its methods. This would allow the method to become more transferable to the other data sources used in this work.

5.3 Future Work

Advancements in relation extraction via machine learning have been made and only continue to improve. If enough data is acquired, this will become a viable tool for this work. This would not, however, be used indiscriminately and would need consult with social network analyzers and individuals in the domain.

5.4 Conclusion

In summary, there is no shortage of research that can be performed regarding case files, dockets, indictments and other court text materials and NLP. There are however tangible improvements that can be made to improve this process or inform a completely different method. Our own method consisted of an intensive document cleaning step, followed by the task of extracting training sentences. A model was then trained to recognize three new labels, "trafficker", "victim" and "hotel". The model was then used on the source data to extract nodes. Following this, the text data is put through a type of information filtering process to obtain arcs between nodes. The outputs are delivered as two separate spreadsheets before being analyzed as networks against human-coded spreadsheets. The takeaway message is that there is no one solution to this research question and that continued advancements in NLP as well as more data can bring this ambition into reality.

Bibliography

[1] D. Kosmas, C. Melander, E. Singerhouse, T. C. Sharkey, K. L. Maass, K. Barrick, and L. Martin, "Generating synthetic but realistic human trafficking networks for modeling disruptions through transdisciplinary and community-based action research," arXiv, 2022, https://arxiv.org/abs/2203.01893.

[2] R. Konrad, "Data science can help us fight human trafficking," Jul 2017. [Online]. Available: https://www.scientificamerican.com/article/data-science-can-help-us-fight-human-trafficking/

[3] J. Tiller and S. Reynolds, "Human trafficking in the emergency department: Improving our response to a vulnerable population," Apr 2020. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7234705/

[4] E. Heil and A. Nichols, "Hot spot trafficking: A theoretical discussion of the potential problems associated with targeted policing and the eradication of sex trafficking in the united states," *Contemporary Justice Review*, vol. 17, p. 421–433, 2014.

[5] C. E., K. Bowers, and L. Vernon, *Using law enforcement data in trafficking research*, 2020.

[6] "Human trafficking 2020," Trafficking in America Task Force, Aug 2020, https://traffickinginamericataskforce.org/human-trafficking-2020.

[7] "Gaps in reporting human trafficking incidents result in significant undercounting," National Institute of Justice, Aug 2020, https://nij.ojp.gov/topics/articles/gaps-reporting-human-trafficking-incidents-result-significant-undercounting.

[8] F. Stokman, "Networks: Social," in *International Encyclopedia of the Social & Behavioral Sciences*, N. J. Smelser and P. B. Baltes, Eds. Oxford: Pergamon, 2001, pp. 10 509–10 514, https://doi.org/10.1016/B0-08-043076-7/01934-3.

[9] R. A. Konrad, A. C. Trapp, T. M. Palmbach, and J. S. Blom, "Overcoming human trafficking via operations research and analytics: Opportunities for methods, models, and applications," Nov 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0377221716308992

[10] P. Szekely, C. A. Knoblock, J. Slepicka, A. Philpot, A. Singh, C. Yin, D. Kapoor, P. Natarajan, D. Marcu, K. Knight, D. Stallard, S. S. Karunamoorthy, R. Bojanapalli, S. Minton, B. Amanatullah, T. Hughes, M. Tamayo, D. Flynt, R. Artiss, S.-F. Chang, T. Chen, G. Hiebel, and L. Ferreira, "Building and using a knowledge graph to combat human trafficking," in *The Semantic Web - ISWC 2015*, M. Arenas, O. Corcho, E. Simperl, M. Strohmaier, M. d'Aquin, K. Srinivas, P. Groth, M. Dumontier, J. Heflin, K. Thirunarayan, and S. Staab, Eds. Cham: Springer International Publishing, 2015, pp. 205–221.

[11] J. N. Lester, Y. Cho, and C. R. Lochmiller, "Learning to do qualitative data analysis: A starting point," *Human Resource Development Review*, vol. 19, no. 1, pp. 94–106, 2020.

[12] A. A. Aronowitz, "Overcoming the Challenges to Accurately Measuring the Phenomenon of Human Trafficking," *Revue internationale de droit pénal*, vol. 81, no. 3-4, pp. 493–511, 2010.

[13] E. Cockbain, *Offender and Victim Networks in Human Trafficking*, 08 2018.

[14] A. Dubrawski, K. Miller, M. Barnes, B. Boecking, and E. Kennedy, "Leveraging publicly available data to discern patterns of human-trafficking activity," *Journal of Human Trafficking*, vol. 1, pp. 65–85, 2015.

[15] M. D. Abram, K. T. Mancini, and R. D. Parker, "Methods to integrate natural language processing into qualitative research," *International Journal of Qualitative Methods*, vol. 19, p. 1609406920984608, 2020.

[16] A. O. of the U.S. Courts on Behalf of the Federal Judiciary, "Criminal cases."

[17] E. Singerhouse.

[18] T. C. Guetterman, T. Chang, M. DeJonckheere, T. Basu, E. Scruggs, and V. V. Vydiswaran, "Augmenting qualitative text analysis with natural language processing: methodological study," *Journal of medical Internet research*, vol. 20, no. 6, p. e9702, 2018.

[19] D. U. Patton, K. McKeown, O. Rambow, and J. Macbeth, "Using natural language processing and qualitative analysis to intervene in gang violence: A collaboration between social work researchers and data scientists," arXiv, 2016, 10.48550/ARXIV.1609.08779.

[20] *Machine Learning and Rule-Based Automated Coding of Qualitative Data*, Pittsburgh, PA, 10/2010 2010.

[21] W. Leeson, A. Resnick, D. Alexander, and J. Rovers, "Natural language processing (nlp) in qualitative public health research: A proof of concept study," *International Journal of Qualitative Methods*, vol. 18, p. 1609406919887021, 2019.

[22] M. Marathe and K. Toyama, "Semi-automated coding for qualitative research: A user-centered inquiry and initial prototypes," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018, pp. 1–12.

[23] A. K. Joshi, "Natural language processing," *Science*, vol. 253, no. 5025, pp. 1242–1249, 1991. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.253.5025.1242

[24] D. Jurafsky and J. H. Martin, "Speech and language processing (3rd ed. draft) dan jurafsky and james h. martin," Dec 2021. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/

[25] M. Honnibal, "Linguistic features · spacy usage documentation," 2016. [Online]. Available: https://spacy.io/usage/linguistic-features#dependency-parse

[26] A. S. e. Infolab, A. Smirnova, e. Infolab, P. C.-M. e. Infolab, P. Cudré-Mauroux, D. of Computer, I. Science, Engineering, and O. M. A. Metrics, "Relation extraction using distant supervision: A survey: Acm computing surveys: Vol 51, no 5," Sep 2019. [Online]. Available: https://dl.acm.org/doi/10.1145/3241741#:~:text=Relation%20extraction%20is%20a%20subtask,structured%20knowledge%20from%20unstructured%20text.

[27] V. Devisree and P. R. Raj, "A hybrid approach to relationship extraction from stories," Jul 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212017316301906

[28] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[29] M. Honnibal, "Entityrecognizer · spacy api documentation," Online, Explosion, 2016-2022, https://spacy.io/api/entityrecognizer.

[30] S. Raschka, "Model evaluation, model selection, and algorithm selection in machine learning," *arXiv preprint arXiv:1811.12808*, 2018.

[31] P. Broniecki, A. Hanchar, and S. J. Mikhaylov, "Data innovation for international development: An overview of natural language processing for qualitative data analysis," arXiv.org perpetual, non-exclusive license, 2017, 10.48550/ARXIV.1709.05563.

[32] H. Alvari, P. Shakarian, and J. K. Snyder, "Semi-supervised learning for detecting human trafficking," *Security Informatics*, vol. 6, no. 1, p. 1, 2017.

[33] E. Tong, A. Zadeh, C. Jones, and L.-P. Morency, "Combating human trafficking with multimodal deep models," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1547–1556.

[34] A. Mensikova and C. A. Mattmann, "Ensemble sentiment analysis to identify human trafficking in web data," in *Workshop on Graph Techniques for Adversarial Activity Analytics (GTA 2018), Marina Del Rey, CA, USA*, 2018, pp. 5–9.

[35] H. Wang, C. Cai, A. Philpot, M. Latonero, E. H. Hovy, and D. Metzler, "Data integration from open internet sources to combat sex trafficking of minors," in *Proceedings of the 13th Annual International Conference on Digital Government Research*, 2012, pp. 246–252.

[36] M. Hultgren, J. Persano, M. E. Jennex, and C. Ornatowski, "Using knowledge management to assist in identifying human sex trafficking," 2016. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7427725

[37] M. Diaz and A. Panangadan, "Natural language-based integration of online review datasets for identification of sex trafficking businesses," in *2020 IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI)*, 2020, pp. 259–264.

[38] P.-L. Huguet Cabot and R. Navigli, "REBEL: Relation extraction by end-to-end language generation," in *Findings of the Association for Computational Linguistics: EMNLP 2021*. Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 2370–2381. [Online]. Available: https://aclanthology.org/2021.findings-emnlp.204

[39] Q. Tan, R. He, L. Bing, and H. T. Ng, "Document-level relation extraction with adaptive focal loss and knowledge distillation," Mar 2022. [Online]. Available: https://arxiv.org/abs/2203.10900

[40] B. Xu, Q. Wang, Y. Lyu, Y. Zhu, and Z. Mao, "Entity structure within and throughout: Modeling mention dependencies for document-level relation extraction." [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/17665

[41] X. Xie and F. Aros-Vera, "An interdependent network interdiction model for disrupting sex trafficking networks," Mar 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1111/poms.13713

[42] B. B. Keskin, G. J. Bott, and N. K. Freeman, "Cracking sex trafficking: Data analysis, pattern recognition, and path prediction," Oct 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1111/poms.13294

[43] E. Cockbain, K. Bowers, and O. Hutt, "Examining the geographies of human trafficking: Methodological challenges in mapping trafficking's complexities and connectivities," Jan 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0143622822000145

[44] E. L. Simonson, "Semi-supervised classification of social media posts: Identifying sex-industry posts to enable better support for those experiencing sex-trafficking," Ph.D. dissertation, Massachusetts Institute of Technology., 2021.

[45] H. Tripp, "All sex workers deserve protection: How fosta/sesta overlooks consensual sex workers in an attempt to protect sex trafficking victims," Nov 2019. [Online]. Available: http://www.pennstatelawreview.org/print-issues/all-sex-workers-deserve-protection-how-fosta-sesta-overlooks-consensual-sex-workers-in-an-attempt-to-protect-sex-

[46] R. Konrad, K. L. Maass, and A. C. Trapp, "A perspective on how to conduct responsible anti-human trafficking research in operations and analytics," Jun 2020. [Online]. Available: https://arxiv.org/abs/2006.16445

[47] T. C. Sharkey, K. L. Maass, Y. Song, K. Barrick, A. Farrell, and L. Martin, "Better together: A transdisciplinary approach to disrupt human trafficking," *ISE Magazine*, p. 34–39, Nov 2021.

[48] T. Le Thi, T. Phan Thi, and T. Quan Thanh, "Machine learning using context vectors for object coreference resolution - computing," Jan 2021. [Online]. Available: https://link.springer.com/article/10.1007/s00607-021-00902-4

[49] K. Clark and C. D. Manning, "Neuralcoref · spacy universe," Online, Explosion, 2015-2016, https://spacy.io/universe/project/neuralcoref.

[50] C. Sammut and G. I. Webb, Eds., *Leave-One-Out Cross-Validation.* Boston, MA: Springer US, 2010, pp. 600–601. [Online]. Available: https://doi.org/10.1007/978-0-387-30164-8_469

[51] F. C. Balato.

[52] n. art.

[53] N. Ben Amor, S. Benferhat, and Z. Elouedi, "Qualitative classification and evaluation in possibilistic decision trees," in *2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No.04CH37542)*, vol. 2, 2004, pp. 653–657 vol.2.

[54] D. L. Hansen, B. Shneiderman, M. A. Smith, and I. Himelboim, "Chapter 6 - calculating and visualizing network metrics," in *Analyzing Social Media Networks with NodeXL (Second Edition)*, second edition ed., D. L. Hansen, B. Shneiderman, M. A. Smith, and I. Himelboim, Eds. Morgan Kaufmann, 2020, pp. 79–94. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128177563000066

[55] Y. Sasaki, "The truth of the f-measure," *Teach Tutor Mater*, 01 2007.

[56] C. Wang, A. Kalyanpur, J. Fan, B. K. Boguraev, and D. C. Gondek, "Relation extraction and scoring in deepqa," *IBM Journal of Research and Development*, vol. 56, no. 3.4, pp. 9:1–9:12, 2012.

[57] T. Liu, K. Wang, B. Chang, and Z. Sui, “A soft-label method for noise-tolerant distantly supervised relation extraction,” in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 1790–1795. [Online]. Available: https://aclanthology.org/D17-1189