Mining Key-Hackers from Large Scale Hacker Forums

Leif Marius Sethne Reppen

02-07-2020

Preface

This master thesis completes the work of two years as a master student in Information security at the Norwegian University of Science and Technology campus Gjøvik. This thesis is written during the spring of 2020, it has been a special spring for everyone. The topic for this thesis is related to digital forensics and machine learning, as the digital forensics track has been my track of choice. I have really enjoyed the two years at NTNU Gjøvik, like the courses, the staff, and the community is something near perfect if you have an interest in information security. It's almost sad it's over. The topic is chosen because of my deep interest in the topic, as well as it gave me opportunities to learn a lot and explore several machine learning methods. This thesis is mainly for those with an interest in digital forensics and machine learning. It's not too complicated to read, but being familiar with machine learning will be beneficial.

Acknowledgment

I would like to sincerely thank my supervisor Professor Vinti Agarwal, who has been with me on this project all along. I'm very thankful for all the ideas and feedback I have received during our time working together. I would also like to thank my supervisor professor Katrin Franke, for stepping out of the shadows now and then, to tell me everything is going to be fine,relax. I'm thankful for the discussion and insights provided. A big thanks to Ph.D. candidate Jan William Johnsen, who gave me access to his preprocessing program and for discussing ideas and problems related to this topic, as well as how academia works. I have to thank my girlfriend Åsne Kristiansen Røsvoll for keeping up with me and for all the support, during these stressful circumstances, it can't have been easy. Finally, I would like to thank my friends and family for all the support, especially towards the end.

Abstract

In today's society, criminals are embracing the opportunities provided by the interconnected world we live in. Organized crimes can be conducted at large-scale online, with only a few keystrokes. Very often with no chance to legally prosecute the offenders. There could be several reasons why it is so easy for malicious actors to get away with illegal activities such as different national laws and jurisdictions, law enforcement capabilities, and lack of cooperation between countries. The Internet is like a paradise for both, low-level criminals as well as serious actors. As cybercrime becomes more relevant than ever, research in the area is booming, however, this is a cat-and-mouse game between law enforcement and cybercriminals. This thesis explores different machine learning algorithms to create a method for identifying Key-hackers from large scale hacker forums. A combination of supervised and unsupervised learning approaches are used to solve the given task. These are: Binary classification using Support Vector Machine (SVM), multi-label classification using Classifier Chains model, and K-means clustering for cluster analysis. The combination of these three models is the pillar in the methodology proposed. Our results confirmed that this approach can easily filter out non-interesting users based on a defined criterion. It can be seen as a good contribution to the digital forensics community and be helpful during a digital forensics investigation process.

Sammendrag

I dagens samfunn omfavner kriminelle mulighetene som internett tilbyr. Organiserte forbrytelser kan gjennomføres i stor skala på nettet, med bare noen få tastetrykk. Svært ofte uten mulighet til å rettsforfølge lovbryterne. Det er flere grunner til at det er så lett for ondsinnede aktører å slippe unna med ulovlige aktivitet. Ulike nasjonale lover og jurisdiksjoner, rettshåndhevelsesevner og mangel på samarbeid mellom land. Internett er som et paradis for både kriminelle på lavt nivå og seriøse aktører. Nå som nettkriminalitet er mer relevant enn noen gang, blomstrer forskning i området, men dette er et katt-og-mus-lek, mellom politi og kriminelle på internett. I denne masteroppgaven undersøkes forskjellige maskinlæringsalgoritmer for å etablere en metode, slik at på identifisering av ondsinnede aktører på internett blir enklere. Tre forskjellige maskin-lærings modeller er brukt for å for å utvikle denne metoden. SVM, Classifier Chains, og K-means- clustering er grunnpilarene i metoden som er testet ut i denne masteroppgaven. Og resultatene bekrefter at denne tilnærmingen lett kan filtrere ut ikke-interessante brukere basert på en et definert standard. Det kan sees på som et godt bidrag til det digitale rettsmedisinske samfunnet og være nyttig under en digital rettsmedisinske undersøkelsesprosess.

Abbreviations

- APT Advanced Persistent threat
- CARVE Collect Analyze, Relate, Validate, Establish
- CNN Conventional Neural Network
- CSV Comma separated values
- DDoS Distributed Denial of Service
- DFI Digital Forensics Investigation
- IRC Internet Relay chat
- ML Machine learning
- SVM Support Vector machine
- SQL Structured Query language
- TF-IDF Term Frequency- Inverse Document Frequency

Contents

Figures

Tables

Code Listings

Chapter 1

Introduction

1.1 Introduction

In today's society, a lot of "classical" criminal activity is taking place online instead of in the real world. For example, skimming equipment found in payment terminals, are reduced from physical devices to only 15 lines of JavaScript [1]. Techniques, scripts, services, and information about potential targets are at large scale exchanged between malicious actors on different hacker forums. In hacker communities, these criminal activities can vary from low-level crimes conducted by script-kiddies to organized hacker groups who can cause severe damage to those who are targeted. Finding the relevant information regarding the key-actors is challenging, yet important in terms of prosecuting criminals for suspect online activity.

1.2 Topics covered by the project

In this thesis, we will work within the areas of natural language processing, feature extraction, and machine learning, to try to identify key actors on hacker forums. We will also use this opportunity to see how machine learning can be used in the digital forensics investigation process.

1.3 Keywords

Machine learning; Natural Language Processing; classification; Feature extraction; Hacker forum; Cyber Crime Investigation.

1.4 Problem description

Underground forums allow criminals to interact, exchange knowledge and trade products as well as services.[2] They play a significant role in executing high-profile cyber-crime activities. A majority of people on these forums are involved,

at most, in minor levels of deviance, thus identification of key actors engaged in serious criminal activity is a complex problem, yet a highly relevant problem to solve for law enforcement. To gather sufficient evidence for a prosecution, solely based on the information found on a hacker forum, is very resource demanding. Thus the need for developing efficient methods to filter out actors of interest is present.

1.5 Justification motivation and benefits

By developing methods that potentially can be used to identify individuals, some real motivation and justification should be put behind. I mean that hunting down malicious actors, will benefit the whole society, by pointing out "what you are doing, is illegal, and we will put measures in place to get you." Will send messages to individuals which could become potential malicious actors in the future. The idea is to come up with a model as can be used to assist law enforcement and/or private sector whenever their goal is to identify a potential threat actor. Machine learning can be very useful to solve complex tasks, where a large amount of data is present and will free up resources if processes done manually, can be automated.

1.6 Research questions

As the main objective is to; Design a model to predict how likely a user will be actor of interest to law enforcement, that can highly benefit the forensic investigation process, some research questions are defined, to keep the project moving forward.

1.6.1 Research question 1

Online forums in general, contains a lot of data, all the messages shared between users, public on non-public posts and metadata about users. Thus, it can be difficult to identify relevant information related to a criminal investigation. For a hacker forum the frequency of posts and messages of interest, shared between users, may be significantly higher than other forums. To identify the relevant information, is challenging, because we need to distinguish between Key-actors and Script kiddies, which with high probability use the same hacker-language. [3]

- How to create a well defined criterion, to separate relevant posts from irrelevant posts?
- How to use the well-defined criterion to distinguish between Key-actors and Script kiddies?

1.6.2 Research question 2

Even though hacker forums provide the “service of private messaging” there is no guarantee that they are not changing their channel of communication.

- How to identify different channels/categories of communication?
- When do actors decide to use another channel to communicate?
- Why do potential users of interest decide to change the channel of communication?

1.6.3 Research question 3

Feature extraction of users is a key element for further analysis of the users in a hacker forum. Features could be used to create a profile of potential actors.

- How can a list of possible attributes/features be extracted from the content of a hacker forum, be used to profile a malicious actor?

In order to solve the aforementioned questions I need access to a dataset which contains relevant information,a computer to conduct the experiments, relevant literature and guidance from my supervisors. I have identified such a dataset, and the dataset to be experimenting on is a dump of the Nulled.IO website. [4]

1.7 Planned contribution

The main objective is to; Design a model to predict how likely a user will be an actor of interest to law enforcement, that can highly benefit the forensic investigation process. Other researchers in the area [5][6][7], have discovered methods to identify the intent behind posts and identification of key-individuals. However, I have yet not found a model which can be applied in a digital forensics process. My understanding is the findings in [5][6][7] are useful resources for my project, but each of them as a standalone research, is not as useful as combined into a model. By the use of previous work and conduct own experiments and analysis and design a model to predict how likely an actor will be of interest, I'm sure the contribution is useful to the digital forensics community.

1.8 Thesis outline

In this section, the organization of the master thesis work written in several chapters is presented. In chapter 2 an explanation of methods and techniques used in this master thesis is presented. chapter 3 explains related work on this topic, some works are closely related with the thesis topic and some are considered as support material. chapter 4 will guide the reader through the process of how we have made decisions and which algorithms are used in order to answer the research questions at hand. In chapter 5 the results of this master thesis is presented. chapter 6 presents our thoughts and reflections up on this work. Finally in chapter 7 we discuss the possible paths forwards as this is just a small contribution, where there are a lot to build upon.

Chapter 2

Background

This chapter aims to provide the reader with basic information about the topics covered in this master thesis. The chapter is split in two, where the first part aim to provide insight without using detailed technical information. While part two, explains the algorithms and models used in this thesis in order to let the reader be able to understand the technical details, that will follow in chapter 4 and 5.

2.1 Theoretical background

This section is providing the reader with theoretical background knowledge related to this topic.

2.1.1 Machine learning in Digital forensics

As digital forensics started to grow as a field, there have been several attempts to automate the process, or at least parts of the process of gathering evidence. However when taking machine learning into consideration of the digital forensics investigation process, it adds a lot of opportunities and capabilities. In this master thesis, we do not apply machine learning in order to extract evidence from disks, networks, or other relevant areas. But we consider developing a model to identify potential criminals on a hacker forum, with the aid of machine learning techniques. In terms of machine learning in digital forensics, figure 2.1 the diagram visualize different areas of a digital investigation. The diagram are provided to give the reader an understanding, that the scope of this thesis, is only a small,but very important area to explore. As the the growth of data are exponential, researching different techniques, to further speed up the process of a digital investigation, is very important, as it cant be done manually. Related research of the area digital forensics, with an emphasis of machine learning and artificial intelligence are further presented in chapter 3.

Machine learning in general is a subfield of AI with three different sub categories. Two of these categories will be briefly explained in the following sections.

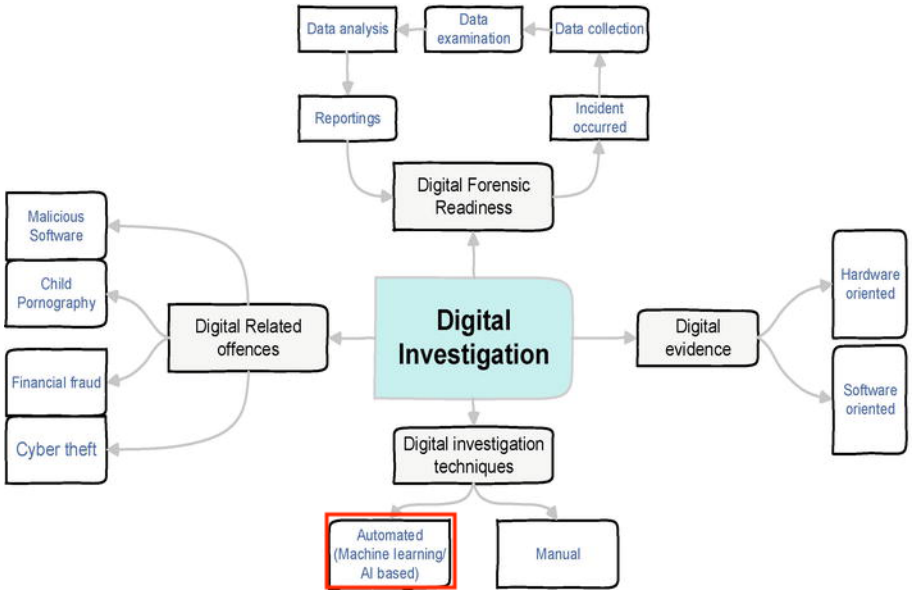


Figure 2.1: Taxonomy

Supervised learning

Supervised machine learning models are models that predict outcomes of different problems based on prior knowledge, a typical problem supervised learning is solving, are the classification and regression problems. In the case of classification, that includes both binary classification and multi-label and multi-class classification problems. In this thesis, both binary classification and multi-label classification are explored and used to solve the asked research questions.

Unsupervised learning

Unsupervised learning is where the algorithm or method in use, do not have any prior knowledge of the input data, thus multiple iterations or epochs might be needed to produce satisfying results. Unsupervised learning is often used when handling clustering problems or dimensionality reduction. In this thesis, clustering is used in the experimental part.

As opposed to supervised learning, unsupervised methods do not require any prior knowledge of the input data. Where supervised machine learning algorithms require labels on training data to make an accurate prediction on raw or unseen data. An explanation of the clustering algorithm used in this thesis will follow in the second part of this chapter.

2.1.2 Hacker Forum

The basis for this whole master thesis is a dump of a hacker forum that was registered to the domain nulled.io up until 6 of may 2016. At this date, the hacker

forum with the slogan, "Expect the unexpected" was breached and dumped on the surface web, for the whole world to see. The forum is still active today, operating under the domain nulled.to. Hacker forums are like a social media platforms, where the users are sharing illicit information, instead of vacation photos. But it can also be seen as a area for learning, as computer interested people sign up in order to learn "how to hack". It is probably safe to say that the users of this forum in general do not have the best intentions. It's not our place to neither judge or define the users based on their internet activities. The information exchanged on such forums can be everything from small scale scamming and bulling of other Internet users, to large scale hacker attacks, or at least planning of conducting severe attacks. As the hacker forum nulled.io which is analyzed in this thesis, it neither stands out as a darkweb-forum selling of intellectual property (IP) and trade secrets, nor does it offer hacker for hire, but in terms of hacking gaming accounts, scamming and tricking malware into unknown victims, nulled.io is such a forum. In the terms of the hacker forum nulled, the majority of users are interesting in hacking gaming accounts, etc. Hacker forums tends to follow a hierarchical structure, where those knowledgeable in terms of cyber security will have an advantage and achieve more reputation, than unskilled users. Essentially a social media platform, where skills will be rewarded by likes(reputation) as of facebook users with pretty vacation photos will be rewarded with likes and comments. An important factor is the sharing of public posts on the forums as well, the higher reputation, the more content and perks the users will eventually gain. Reputation can be achieved of being active on the forum, such as contribute with public forum posts seen relevant, donate money to the forum or buy VIP access. The more reputation a user get, access to "higher ranked groups" and access to "hidden-content" can be be provided.

2.1.3 Key actors

As explained in the aforementioned subsection, a hacker-forum has some sort of hierarchical structure. Usually where knowledgeable users and users who share their knowledge are rewarded. When we talk about Key-actors in hacker forums and key-actors operating for a nation state, we need to differentiate between these groups. As a hacker forum might be the starting point for a career in cyber crime, it is my understanding that a hacker-forum that can be accessed from the surface-web, will not hide nation state actors. Thus we need to establish an understanding of who we are looking for and why. When reading about hackers in 2020, they are usually put into three different groups, nation state actors, hacking as a service or hacking for finance, and hacktivists.

2.1.4 Nation state actors

Nation-state actors are often referred to as APT-Advanced persistent threat, they might not have direct ties to a government, but indirectly, they are often associated with states known for taking easy on human rights. These different groups

are closely tracked by major security firms such as Mandiant (FireEye) and Crowdstrike[8], and probably by governments that are considered targets for APT-groups if they possess this capability. As APT-groups are considered professional hackers that might act on orders from governments around the world, the probability of identifying these type of hackers in the hacker forum we are using for this thesis, is very unrealistic. However, they cause a severe threat to society and need to be mentioned.

Financial motivated hackers

There are also groups as referred to as financial motivated groups[8], that may or may not work for governments, but the majority of their interests is to earn their money by conducting illegal cyber crime activities. As they will potentially do everything from stealing credit-card information, install banking Trojans on victims computers to phone scamming, the skill level of these groups can vary from expert to novice. However these are the type of hackers that might be possible to identify in a hacker forum we are analysing in this master thesis.

2.1.5 Hacktivists

Another actor that we have mentioned is hacktivists, hacktivists are those who use their knowledge within cyberspace to cause damage for a cause they believe in.[8] Probably the number one famous hacktivist group in the world is known as Anonymous.[9] It's my understanding that there is a probability of finding hacktivists in this kind of hacker forum.

2.1.6 Hackers Expected to operate from nulled.io

The last group of hackers which I can consider finding relevant in the data we have been exploring, are those who commit small crimes for fame and reputation. Not unlike "Instagram influencers" today, except half-naked photos are swapping place by conducting Distributed Denial of Service attacks, (DDOS) online credit-card-skimming, hacking of gaming accounts, Netflix accounts, and mail accounts. To gain reputation within their hacker community. These hackers will often be categorized ~~as script kiddies~~. Within a hacker community, the most interesting users are not those who hack for fame nor a cause. Both those actors who can stand by and pull the strings to get others to do the job, if those users exist at nulled.io is left to see.

2.1.7 The Digital Forensics investigation process

To explain the digital forensics investigation process, a definition of the Digital Forensic Science is provided; *The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources*

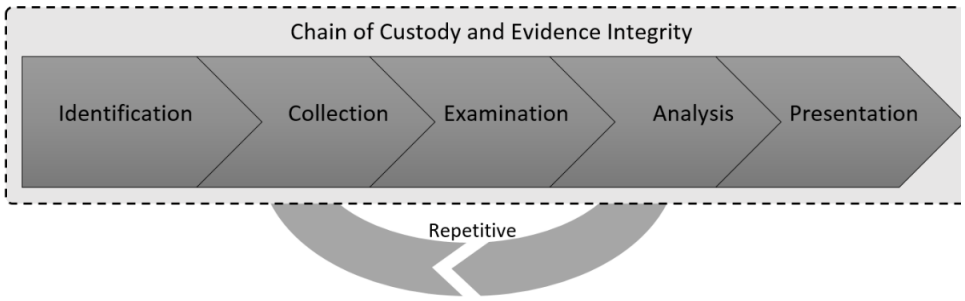


Figure 2.2: Digital Forensics Investigation process

for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations.[10]

When investigating a criminal act related to computers, hard disks, phones, networks, tables, or anything related to information security. The digital forensics investigation process is inevitable not to mention. To explain how the work is conducted in this thesis, a simple explanation of what digital forensics is and how the process can be used in light of this work is at is place. The digital forensics process includes five steps as mentioned in [11]. Figure 2.2 shows the five steps of the DFI-process, however as this process is maintaining evidence integrity, it is not the reason to bring it up in this setting, but as a framework to show that this theoretical model actually can be applied in this project.

As Figure 2.2 shows this process contains repetitive steps. The repetitive steps of the DFI-process are the most relevant for this master thesis, as this is a practical approach including collection, examination, and analysis, presenting the outcome in court is not an issue. As in this master thesis, no seized data has been made, nor is the need to maintain the integrity of evidence. However, as this is a thesis aiming to either approve one method or argue that the method applied is not satisfying. Using the digital forensics investigation process to explain some of the steps taken, is considered relevant.

2.1.8 Retrieving data

When looking for data to explore in terms of exposing a hacker forum, related work shows that crawling online forums to gather public posts with the help of a web-crawler is usually the way it is done. In this thesis, the data is found as a 10Gb SQL database online. In light of the digital forensics investigation process, Identifying the data to use in this thesis was the first step. And the data identified was nulled.io hacker forum. The next step is to collect and identify the data needed from the database, as the database is huge and contains a lot of data that goes beyond the scope of this thesis to explore. As mention earlier scraping public messages of the content found available on hacker forums, creates the need of using a web-crawler. As we already collected the whole database this is not needed

in our case. However, we did not analyze the public posts of the forum, as this has already been achieved to some degree by others, explained further in related work. As we have access to the whole database, we took the liberty to only use the private messages sent between users, to hopefully gain new insights into secrets shared in a hacker community.

The database consists of 211 tables, however, 74 of these tables appear to be empty. The tables which hold information about private and public communication, are the tables I have used mainly for my experimental part. To explore and identify the whole structure of the database and name all tables, are however out of the scope of the thesis. In addition to the tables consisting of private one-to-one communication and the tables of public forum posts, some tables help understand the internal structure of the forum, for example, a table named "groups" this table provides information on which group each member are assigned to. There are 20 different groups, however, only 14 has members assigned to it, by a group id.

2.1.9 Data Cleaning and preprocessing

In general, when handling online forums, the data retrieved from the internet contains a lot of noise. This can be but is not limited to include HTML-tags and URLs, bitcoin-addresses, images, emoticons. As a second step of the collection phase, we need to capture the exact data and make it useful for further examination. When dealing with a machine learning problem, we want the data to be readable for a machine not only humans. The techniques used to prepare the data is often referred to as data cleaning and preprocessing. The cleaning process aims to get rid of the data that can make the dataset noisy. This includes removing HTML-tags, URLs, image links, and so on. When the dataset is stripped for this noise, it is less noisy, but it still contains words that are not considered useful for computers. Thus standard steps to further improve and prepare the dataset are usually conducted. The reason to accomplish such tasks is to create the best possible features, both to achieve better results, but also to reduce training time when applying machine learning models, as the nulled.io dataset is huge. There are a couple of other techniques that are very useful when handling text, known as stopword removal, stemming, and lemmatization.

Stopwords

Removing stopwords, is the process of removing words such as "a", "the", "and", "I" and so on, all words which are necessary for a sentence to make sense, to humans, while it will cause noise to computers as extra unnecessary features. Usually for each dataset that includes text processing, a new word list, needs to be generated to get rid of custom for e.g, a hacker forum. How this process was accomplished is further explained in chapter 4.

(a) For a linear two-class classification problem there exist several hyperplanes [12]

(b) Support vector machine Optimal hyperplane[12]

Figure 2.3: Support Vector Machine

2.2 Technical background

This section aims to provide a brief technical explanation about the machine learning models used in this thesis. Also, some alternative techniques which are not used in the thesis, but still explored and experimented with, are explained in the following subsections.

2.2.1 Binary classification

Binary classification can be explained as simple as; "is this true or not." As one of the major parts of this thesis, an explanation of one of the most common machine learning classifiers, which was used in this thesis is explained here. We tested three different classifiers, all provided good results, but as the goal of this thesis is not to compare different machine learning methods, we will explain the one we proceeded with; Support Vector Machine- SVM.

Support Vector Machine-SVM

Support vector machines are one of the most successful machine learning methods developed for solving classification and regression problems[12]. SVM is using all available attributes in order to make a prediction to a target variable. The attributes are used in a linear combination to do so. And this make SVM very useful for large datasets and is one of the reasons it is used here. "The basic idea of SVM methods is to place an optimal class separating hyperplane in the space of original attributes. If the learning examples are linearly separable, then in general there exist several possible separating hyperplanes"[12] as can be seen in Figure 2.3a.

An optimal hyperplane is equally distant from the nearest samples for both classes.Figure 2.3b The earning examples nearest to the optimal hyperplane are called support vectors. And the distance between the support vectors and the hyperplane is called a margin. An thus if the distance between the support vectors and the margin is maximized we have an optimal hyper plane[12].

2.2.2 Multi-label classifiers

In the category of supervised learning, where we find classification as one out of two subproblems, we can categorize the classification problems into binary classification as described above, multi-class classification and multi-label classification. Multi-class address the problem of predicting which class the data belongs to, even when a more complex problem than yes or no, are present as there are several different classes present.

When looking at multi-label classification problems we can find that several labels exist in different classes at the same time. Dealing with textual problems, this is a relatable issue, as a comment or a message, can be categorized with several labels. Consider the sentence; "At 6 pm we are going to launch a cyberattack, against a power plant, all coordination and information that will follow, will from now on be discussed at discord" As for binary classification, we can ask; is this it-security related material; yes or no? We can by adding multiple labels ask; does this sentence contain information about; time, it-security, communication channel, target, and users? The answer; is, yes, yes, yes, yes, no. By using multi-label classification we can extract more information out of a sentence than; yes or no. However creating good labels to produce good results, can be very tricky. To solve this problem there are three major referenced techniques shown in Figure 2.4.

Problem Transformation: Transform the problem into a binary classification problem, such as each label will be treated as one binary class.

Several different algorithms and models are used to solve this issue, some of the most common methods known are: Binary-relevance,classifier chains,

Adapted Algorithm: Adapting the algorithm to directly perform multi-label classification, rather than transforming the problem into different subsets of problems. Some of the well known Adaption algorithms are; K-nearest neighbour, Decision trees, Kernal vectors, and neural networks.

Ensemble approaches: An approach where a multi-class classifier is used to solve a multi-label classification problem. Some of the most common methods are Label Space Partitioning Classifier and Majority Voting Classifier.

Classifier chains

Classifier chains is a multi-label classification method as belong to the problem transformation category. Classifier chains combines the efficiency of the binary relevance method, while still taking all the relevant label dependency into consideration for making a prediction. In other words, the feature space is expanded by all the labels, before they are clustered into groups and forming a "classification chain of each label [13]. Classifier chains builds upon the principle of logistic regression[14].

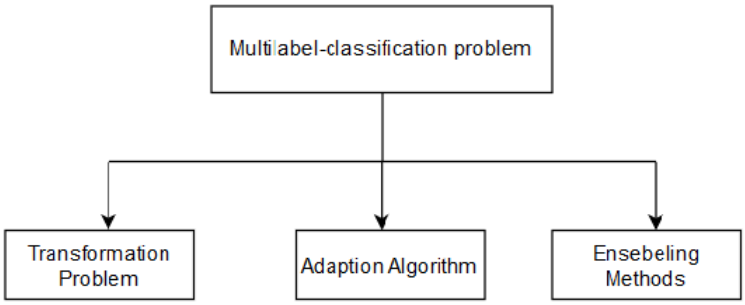


Figure 2.4: Multi-Label-classification approaches

2.2.3 Clustering

Clustering is an unsupervised machine learning technique that is used as the final machine learning step in this thesis. In the following subsections, a brief outline of the two main clustering techniques, partial clustering, and hierarchical clustering, is given, followed by the explanation of K-means which is the algorithm used in this thesis. In this thesis, the partial clustering plays a part towards the end of the experimental part. As one of the most applied unsupervised machine learning methods used, clustering is applied in the digital forensics science. As an example where the text is a part of the evidence, and clustering analysis is used to explore and classify evidence [15].

Hierarchical clustering

In hierarchical clustering, each example initially forms a separate cluster. At each step of the algorithm, the most similar clusters are amalgamated, thus forming a tree of clusterings (dendrogram). Frequently, pairwise amalgamation is used, leading to binary trees. Amalgamation continues until all examples belong to a single cluster. Finally, the learning algorithm or the end user, selects the most suitable clustering level from the constructed tree. In divisive (top-down) hierarchical clustering, all examples initially belong to a single cluster. The number of clusters is incremented at each step of the algorithm by dividing an existing cluster into (usually, two) sub-clusters. The process continues until a suitable number of clusters is obtained.[1] "[12]

Partitional clustering.

Initially, the number of required disjoint clusters needs to be known. Given a criterion for measuring adequacy of a partition of examples into c clusters, the algorithm searches for optimal partitions of examples. The process starts with an initial clustering, frequently obtained by randomly selecting c examples, and a set

[1] This part is a an excerpt from[12] as the explanation of hierarchical clustering is well explained.

of transformations for changing a partition into another partition. The learning algorithm modifies a partition until no allowable transformation would improve the given criterion of partition adequacy.[2] " [12]

K-Means

K-means is one of the most popular partitional clustering algorithms in use today. And it is also one of the simplest and fastest clustering algorithms. As the algorithm is using squared euclidean distance as the optimization criteria, it is to be defined as an square error algorithm. The basic structure of the algorithm can be found as pseudo code for the K-means algorithm in algorithm 1. As the main advantages of k-means are speed and simplicity, hence it works very well on large datasets, which makes it very useful for our case. However all the attributes used as input to K-means needs to be continuous values in order for the algorithm to work [12].

Algorithm 1: Pseudo code of K-means

INPUT: A set of learning examples to be clustered, and the number k of desired clusters.
OUTPUT: Partition of learning examples into k- clusters.
Randomly generate K clusters and determine the cluster centers or directly generate k seed examples a s cluster centers
repeatAssign each example to the nearest cluster center Recompute the new cluster centers.

until no example has changed from one cluster to another

2.2.4 Turning words into numerical values

As a computer doesn't appreciate text the same way we humans do, converting the text into numerical features is necessary while handling the text classification problems. There are different ways to accomplish such tasks, and the next section will briefly explain two of these approaches.

TF-IDF

TF-IDF is a statistical approach, calculated by using the term frequency of a document and the inverse document frequency. It has its origin from [16]. The matematical formulas are described in Figure 2.5. This is one of the standard methods in order to convert text to numerical values. TF-IDF is an statistical way of weighting the importance of words in a document. In our case, every private message is considered as a document. The overall advantages of this approach are speed and efficiency, while a major disadvantage is that TF-IDF can not capture the context of a sentence. However we justify using TF-IDF because of its efficiency.

[2] This part is a an excerpt from[12] as the explanation of partitional clustering is well explained.

$$tf(t,d) = log(1 + freq(t,d))$$

$$idf(t,D) = log(\frac{N}{count(d \in D : t \in d)})$$

$$tfidf(t,d,D) = tf(t,d) \cdot idf(t,D)$$

Figure 2.5: Formula for Calculating TF&IDF and TF-IDF

Word Embeddings with Word2Vec

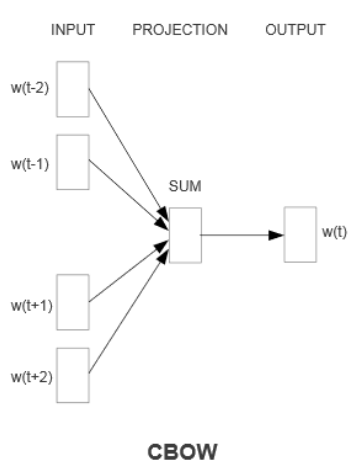
Word2vec[17] is a model used for word embeddings as a two-layer neural network, it is a quite powerful unsupervised machine-learning algorithm. The model was developed in 2013 by google [17] and comes with two different architectures. One which is predicting words based on a context and one predicting a context based on a word. respectively called continuous bag of words(COWB) and Skip-Gram. In this case word2vec is simply used for transforming text into vectors, in order to be able to classify the dataset further on. Word2vec turns out to be very useful when the goal is to keep the context of the words in a document, but still translate the words to vectors, known as word embeddings. A little less naive than TF-IDF, which defines words who appear frequently as more valuable and thus assigns a higher vector value, despite not being relevant towards the context. However one disadvantage of creating word2 embeddings with word2vec is the training time of the model. There exists pre-trained models which can be used for this purpose. To be able to create the best possible word embeddings to extract the most relevant features, it should be trained on the particular dataset. As we have experimented with this, training time took about 3 hours on a dataset containing 500 000 messages.

continues-bag of word-model

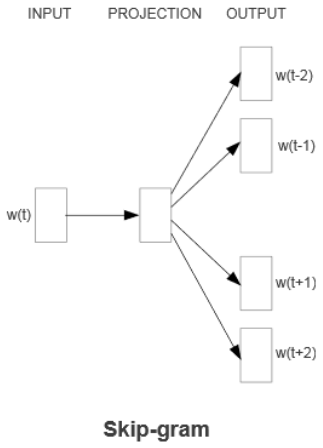
The COBW can be explained by Figure 2.6a, essentially the idea behind COBW is to predict a target word, based on a context window. As an unsupervised machine learning algorithm, the model is just fed the information and it will construct word embedding from the given vocabulary created by the dataset of use.

Skip-Gram-model

The skip-gram model on the other hand tries to predict the context, based on a target word. It is easier to understand the concept by looking at Figure2.6b. By



(a) Continues Bag Of Words Architecture [18]



(b) Continues Skip-Gram Architecture [18]

Figure 2.6: Word2Vec architecture

one target-word as input word, it predicts the probable context. But can also be used to compare similarities between other words.

Chapter 3

Related work

This chapter contains research from related topics to this master thesis. As the goal of developing a model that can benefit the DFI-process it is important to look into a couple of other related models which already are proposed.

3.1 Theoretical models

When first addressing the problem of creating a model that can benefit the DFI-process, it is important to understand the DFI-process on its own, and why there exists such a process. The simple answer is, since Digital forensics is a relatively new field, and there exists a gap between the legal system and new technology and many pitfalls to be avoided for evidence to be admissible in court[19]. Therefor a proven concept such as the DFI-process needs to be in place, to maintain the integrity of both the investigator and the evidence presented. As explained in [19] the forensics investigator needs to address all aspects, both the technical but also legal frameworks, to make this model work. And it's worth noting when we are trying to apply machine learning techniques to solve a crime.

An area where recently a lot of research is conducted are so-called threat hunting, threat hunting is a countermeasure in cybersecurity, aiming to actively defend valuable assets from perpetrators. As the DFI-process, the framework proposed by Sqrrl [20] is also an iterative process, and when actively conducting threat hunting the need of involving Digital Forensics techniques are present [21]. However, some differences exist between a threat hunting process and the DFI-process where the DFI-process main goal is to collect evidence. However, both models are hypothesis-driven. One scientific model proposal named CARVE [22]- (Collect, Analyze, Relate, Validate, and Establish) can be seen in relevance with this project, as the CARVE model addresses Threat Actors as on of seven concepts which need to be established to create a solid threat hunting hypothesis. Where threat actors and Key-actors, are considered the same, but from different domains, in this scenario, a real threat actor can also be classified as an actor operating from a nation-state. While the Key-actors we are looking for in this thesis are higher-ranking members of a hacker forum. When narrow down to the technical details,

there will be some similarities, while the frameworks and initial goals are different.

As hacker forums can be seen as a deviation of social media platforms, it is still an online platform, where information is exchanged. This is why we point out this the work of "Towards Designing a knowledge Graph-based Framework for Investigating and preventing crime on Online social networks"[23]. As this contribution points out that many frameworks are created to solve some specific tasks, such as detection of malicious actions, prevention, and visual analysis. However, very few frameworks directly address the issue with the collection of Digital Evidence. However in [23] different challenges are identified; including developing a model to organize and integrate the massive volume of different data-types in online social networks, develop storage technology for storage of the data collected in challenge one, automatically extraction of content, develop a model for the preservation of digital evidence and finally developing appropriate methods to acquire data from social networks. In theory, these are great ideas, while as stated in [24]. The more complex the framework is, thus more of its specialization gets lost. As graph-based technology can be efficient it requires a lot of storage, however, the steps proposes, would be considered more feasible if they dropped the knowledge based-graph. This master thesis is trying to solve a similar problem, just narrowed down to a hacker forum

3.2 Research questions

As we have presented different theoretical approaches related to this master thesis, we would further address research related to the specific research questions, as they are a guideline for how to carry out the proposed model. There is simply no solution fits all, a lot of methods and techniques are tested out for different purposes. However, we have identified many of the following techniques and methods as useful resources in terms of completing this thesis. A fair amount of papers have been read to get an understanding of how to approach the research problems described and further on if it is possible to use some of the earlier proposed techniques. When conducting a literature review for this project as an early step it is most recent research as stand out, specific to the topic extracting information from a hacker forum. One of the first articles that were discovered was a research paper dated back no later than 2016 [25], as directly addresses the extraction of content and analysis of hacker forums. Further on very much of the relevant literature is discovered to be published in the past few years. Some of the papers address the research questions directly and have solved an issue or proved how machine learning algorithms can be very useful when applied with well-crafted features as input. while some of the research articles cover broader than one question and some only address one of the questions.

3.2.1 Distinguish between relevant and irrelevant posts

1. How to create a well defined criterion, to separate relevant posts from irrelevant posts?
2. How to use the well-defined criterion to distinguish between Key-actors and Script kiddies?

The first question has to a certain degree been answered in [26], where the authors have labeled posts from a hacker forum as classes of relevant and irrelevant posts. However the article compares classification methods, and those not go into an in-depth analysis of the actual forum post and how to extract cyber threat intelligence. However, the criteria made for separation of relevant classes is sound, and stand out as a solid starting point. The approach used in [26] is focusing on the public forum post, while we decided we should do our experiments on the private forum messages. Essentially the binary classification problem raised in both topics, are similar to some modifications. The topic neither addresses to question of how to separate different actors on a hacker forum. In [26] sections of future work, the authors point out the importance of analyzing posts considered relevant, as well as the need to explore other algorithms and methods than; Support vector machine(SVM)and a Conventional Neural Network (CNN).

On the other hand, the article from 2018, "Mining Key-Hackers on Darkweb Forums" [7], addresses the same question as we are doing here. However, the authors are using a very different approach to explore their problems. And as we are asking the same questions, it is important to explore other options as well. As their approach is to extract 25 different features to describe the users of a Darkweb forum, we rather create these features later, based on criteria retrieved along the way. The features described in [7] aim to describe a Key-actor, where things like several public posts, length of posts, frequency of posting, technical writing, and so on. When all these features are gathered, they apply the genetic algorithm to filter out the least interesting users, to be left with the Key-actors. A very elegant approach, however, it's a different direction of where we are heading, while both aiming for the same goal.

A third research paper indirectly addressing these questions is: Identify Central individuals in organized criminal groups and underground marketplaces [6]. Here the authors are using centrality measures to identify key-actors on Nulled.io, the very same forum as we are exploring. The main difference here, the approach to identify users, as centrality measures are used as the approach to identifying individuals. Centrality measures will reward a large number of messages or public forum post for each user, meaning the very active individuals, will turn out as the interesting individuals of this forum. luckily for us, the article distinguishes between public forum post and private messages, and as we are exploring the private messages, we can compare our results, to the results provided in the article [6]. As the authors conclude, the users identified, can be categorized as script kiddies, this could be an indication if our model fails or not.

3.2.2 Identify communication channels

1. How to identify different channels/categories of communication?
2. When do actors decide to use another channel to communicate?
3. Why do potential users of interest decide to change the channel of communication?

All of these questions are relatable to the first section of questions if we acknowledged the fact that we have access to a whole hacker forum. One of the research articles which addresses the question related to communication is the article "Exploring threats and vulnerabilities in hacker web: Forums, IRC, and carding shops" [27]. They specifically point out that users of a hacker forum are communicating over IRC-Internet Relay Chat, and create a method to try to intercept these IRC channels. We are not looking to intercept IRC channels in this thesis, however, being aware that users tend to change channels of communication is an issue, need to be kept in mind while moving forward with this project.

Another article that comes in handy is 'Automatically identifying the function and intent of posts in underground forums [28]. As the authors are creating different categories related to different topics to understand the intent of these forum posts. As we are creating categories to label the private messages, the approach proposed in [28], can be used to answer the aforementioned questions. By combining the different approaches proposed in [28] [5] [26], as they all are quite relevant addressing the topic of hacker forums, the questions asked can probably be answered.

Another article that provides a lot of useful insights in terms of understanding the hacker culture is "The Competing Values of Hackers: The Culture Profile that Spawned the Computer Revolution" [29]. As this article provides information, over a 25-year long timeline, from the very famous hacker forum "The cult of the dead Cow." It provides insights at least in the light of the "Competing values framework" used by the others to create four different cultural archetypes, based on the communication between hackers. As this article give an interesting view and in-depth insight into the structure of a hacker forum, it does not directly address the research questions mention, but rather provides information, on how to understand the culture and communication in such a forum. After all, it seems to be rather few articles to exist addressing the questions related to communication, as most articles found, are aiming to find key hackers, extract cyber threat intelligence or understanding the hacker culture.

3.2.3 Feature extraction

This question regarding feature extraction of a content, is mainly a technical question. As we deal with a text problem, referring to articles using feature extraction, or vectorization, should cover it.

1. how can a list of possible attributes/features be extracted from the content of a hacker forum, be used to profile a malicious actor?

Most of the articles listed for the previous questions can be of help when looking for how to extract features from documents, but there is one old yet very important technique described in [16] called TF-IDF which needs to be described when handling text classification problems. However, there is a dedicated subsection in chapter 2 on how TF-IDF works. But as relevant literature, the discovery of TF-IDF as is a very important way of weighting terms in a text-document before applying machine learning algorithms, it deserves to be listed here. There exists also a new modern approach called word embeddings, as can be extracted by different algorithms, one of the approaches also described in chapter 2 is word2vec [17]. As several different algorithms can create word embeddings, we can't describe them all. If we look at the question asked and do not look at the technical side, where we turn text into vectors, the article already described related to research question one, the extraction of 25 features in [7]. Is a good example of what features can be extracted and how to apply them. We would also like to address the article 'Automatically identifying the function and intent of posts in underground forums [28]. As a good basis for creating features and to further build upon.

Chapter 4

Methodology

This chapter aims to provide insights about the methods used to answer the earlier formulated research questions. It is important to have good understanding on the choice of methodology that is best suited to our problem. I spent some time on deciding what are the right methods to investigate the research questions proposed in this thesis. The structure of chapter goes like this: First a brief discussion on research methodology is presented, Then the description of entire workflow process is given (Figure 4.3). It has several stages included: Data extraction, Data preprocessing, Binary and Multi-class classification of hack forum posts, User profiling (n-dimensional vectors) and clustering of users to identify persons of interest on the hacker forum.

4.1 Research Methodology

When choosing the appropriate method to use for this type of study, the first step was to search for relevant literature. In the two books Research Methods for Cyber security[30][31] and Practical Research-Planning and Design[32], I found guidance on how to choose the most applicable methodology. When deciding the research method, I spent time on identifying and analyzing my available resources such as literature, research questions and the dataset to conduct experiments with. I identified that a descriptive research method is the most applicable for my task. In the book practical research for Cyber Security[30], descriptive research is broken down into four different areas as shown in Figure 4.1. These areas are: Observational, Experimental, Theoretical and Applied research methods.
By definition the data was already collected before starting on this project. The research questions at hand, was best fitted to be answered by conducting an observational study, where the behavior of the users on a hacker forum is the elements to be analyzed and observed. From [30], a few of my questions can be addressed by making predictions with machine learning. I will further explain each of these questions to highlight how the selected research method applies. I will also explain each of the steps followed in the light of the digital forensics investigation

process, as of the goal of this thesis is to develop a model that can highly benefit the digital forensics investigation process.

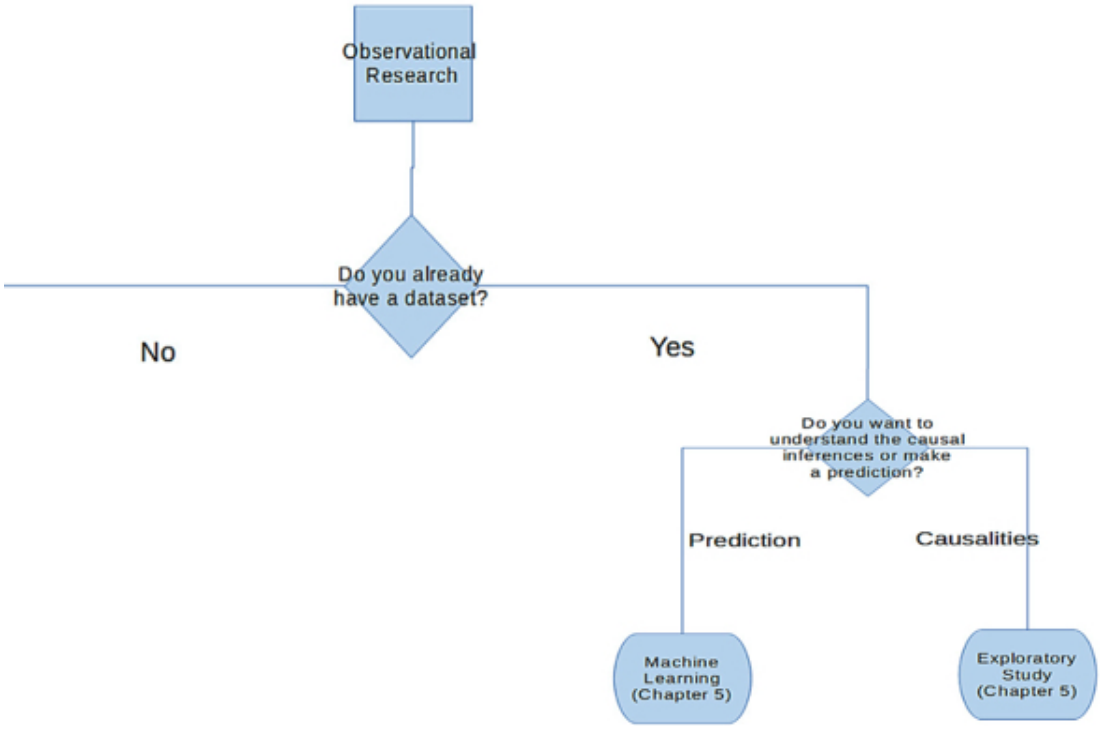


Figure 4.1: Flow Chart- Choosing method

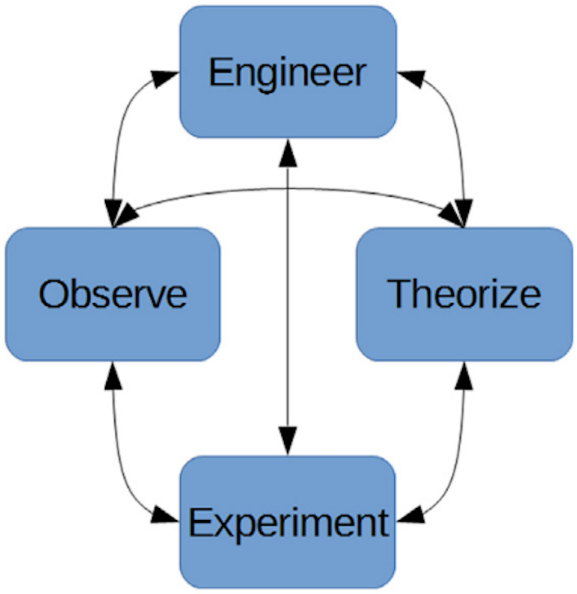


Figure 4.2: Methodology [31]

When working with a dataset of the huge size, it is not possible to get all right answers at once just by choosing the appropriate methodology. It is close to impossible to follow a "waterfall-approach", when the project at hand includes machine learning and a iterative process such as the Digital Forensics investigation process. Following the whole project with a waterfall approach, where first a literature review is conducted, then the experimental part, and at the end finalize the writing, would in my opinion not justify the way of working, as all these areas needs to be revisited multiple times. The methodology follows an experimental research approach, several of the areas have been revisited multiple times. Figure 4.2 explains at my experience of conducting this research. Both as a way to complete the whole project, but especially the experimental part, including programming, where documenting parts of the experiments, as well as observe results in order to go back and engineer new features, have been crucial in order to progress.

4.2 The Workflow Process

This section explains how each of the components of this thesis have been accomplished. To better understand the workflow, a block diagram is created as shown in Figure 4.3. It gives a visual impression of how exactly the steps are followed. Details of each of these steps are described in following sections.

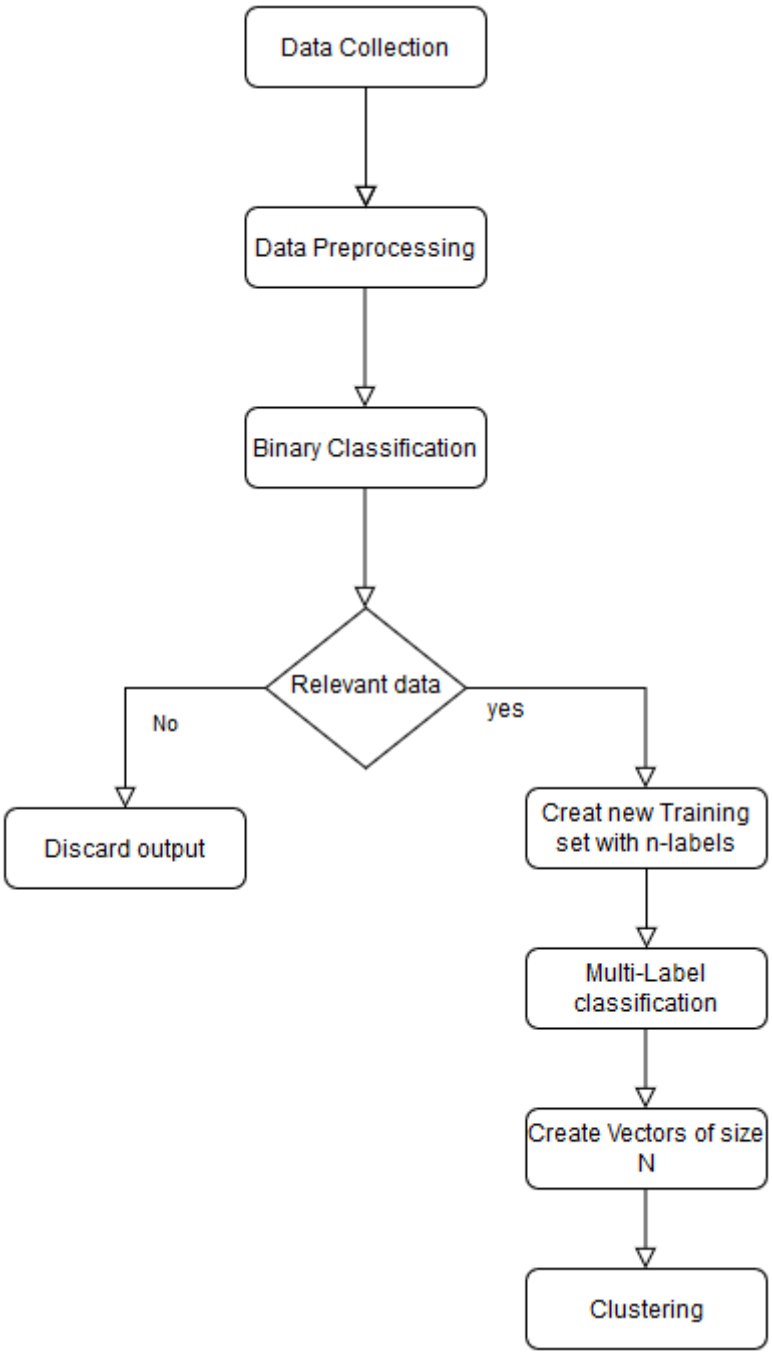


Figure 4.3: The work process, of this master thesis

4.3 Data Collection

As mentioned earlier the database was collected from a website [1] and downloaded as a 10 Gb SQL-relation database. The database was then imported into a my-SQL server and accessed by using python (An example in shown in Appendix A), as the programming language supported with pandas[33] python library. The tables extracted from the database were initially extracted using SQL-queries to get the needed data from specific tables, then the tables was stored into pandas data frames before it was saved as a csv-file. The description of some interesting tables is given below.

4.3.1 Members

During the collection phase, different queries were executed to extract different data in order to understand the structure of the hacker forum. The columns extracted during the collection phase from the table "members" was msg_author_id, msg_post and name. These tables contains the unique user ID of each member and all the private messages sent by each member as well as the users display name. This data create the foundation of the dataset to conduct experiments on.

4.3.2 Groups and Emoticons

Two other interesting tables in the dataset are: groups and emoticons. The table **groups**, holds the information about which group each user belongs to, and could be a starting point of which users are more likely to be of interest than others. But this was not taken into consideration at first. As we got more familiar with the hacker forum, we discovered during preprosessing and manually labeling of relevant data, that some particular users who belong to certain "interesting groups" had a username which was equal to the command of an emoticon. This could indicate that these users are more likely to be vital users of the hacker forum, but this lead was not pursued further at this point. When the tables, counting the private messages and the unique user-ID, was saved to file, we decided that the user with msg_author_id= 0 and msg_author_id = 1, should be dropped. These two users are clearly administrators, and was sending out informative messages to all users, such as "Welcome to nulled.Io username", and by dropping these to users, we could reduce the dataset between 200,000 and 300,000 messages. Which reduced the time to clean the dataset as explained in the next step.

The SQL-table **Emoticon** was identified as typing commands for emoticons. The list of emoticons has a length of 133, meaning 133 words and combination of signs can potentially be removed. But after closer inspection, it turned out that a couple of the words or commands to produce emoticons, was also recognized as usernames of other "respectable" users which could be an indication of interesting targets.

[1] `https://archive.org/details/nulled.io_database_dump_06052016`

The choice of using a four year old dataset, is justified by the fact that a complete dump of a hacker-forum, is quite rare. Also the time saved from not using a crawler to gather data, as well as the possible complications in terms of the legal and ethical aspect of web-crawling, was considered. Even some research has been conducted on the nulled.io forum already, there is still much to explore and understand. As we are developing a model which can benefit the forensics investigation process, experiments on data that can be related to seized evidence, is a valid point to justify this decision.

4.4 Preprocessing

When the previous step was completed, the next process in line was preprocessing step. As we have a text classification problem, we need to turn text into numbers, readable for the machine. In order to do so, it is essential that we get rid of as much noise from the dataset produced as possible. Such noise are html-tags, URLs, numbers, white-spaces, and tabs, punctuations, and special characters. Further the removal of stopwords and lemetizations and tokenizing the text is a part of the preprocessing process. As well as username, email and password-dumps which is stripped from the datasets in order to make the datasets ready for further analysis. In the dataset we identified "words" which was not possible to explain, such as 'Kappa', Kappahd, 'Kappaross' ,'fappa', as emoticons command. However, to deal with the problem of missing relevant usernames, a small script was written to search in the list of usernames, and compared with the emoticon table. Then the relevant usernames was kept, and the other commands was included in a customized list of stopwords and then removed.

4.4.1 Stopwords

As a part of the preprocessing stopwords was removed, in order to reduce the datasets complexity and get rid of words which usually won't add any value to the sentences. As we are using TF-IDF to vectorize the private messages, we won't capture the meaning in the sentences produced, thus the need of including words such as "the", "a" "and" "an" on so on is not present. This was accomplished by using a custom made stopword list, dedicated to the nulled.io hacker-forum. The list containing the stopwords was then given as input for a python script, handling all different preprocessing tasks.

4.4.2 Creating training dataset

When the initial preprocessing phase was completed, a training dataset to be used for binary classification was created. This was saved as a csv-file only containing the unique user-ID and the private message sent from each user, before it was labeled with semi-manually approach as explained in the next sub-section.

Table 4.1: Representation of relevant- irrelevant

msg_author_id	msg_post	Relevant
1239	deactivate access scan active virus scan antivirus	True
879	lolololololololol hahahaha	False

4.4.3 Criteria for labeling private messages

To be able to separate the relevant from irrelevant posts, a naive approach was used together with manual inspection. As a part of preparing the dataset for a binary classifier, a small script was written containing relevant keywords related to security and communication.

The argument to speed up the process with a naive approach such as using keywords is based on previous work on the same hacker forum [26]. As we have in this project focused on the private messages extracted from the dataset, the list of keywords used for automating labelling was slightly different. Together with manual inspection and the automation of creating labels, the dataset created was considered sound enough to filter out relevant data from irrelevant data. For a private message to be considered relevant, it needed to contain keywords related to security or communication platforms, as well as other online forums. As manual inspections show that some of the users are continuing their private conversations on other platforms outside of this hacker forum. Some of the communication platforms considered to be relevant are; Skype, Facebook, WhatsApp, Telegram, discord, Keybase and signal. As some of these platforms can offer encryption capabilities. In terms of relevant security keywords, some of the keywords used are; exploit, hacking, hacker, malware, infection, infect, SQL, injection, and keylogger, in different variations of the words.

The dataset created based on this criteria was build on 4000 selected private messages, to further be able to extract 10000 features and build vocabulary, to both train the binary classifier and to use the vocabulary on the raw data, to predict relevant data on the remaining dataset, not used for training. Each message of the users was iterated through and compared to the list of related words. If a word was present in the list of relevant words and a present in a message of a user, the Boolean value "True" was return and saved in a pandas column "relevant". If there was no match, "False" was returned. Then finally, all the messages were saved in a CSV-file with the information: *unique user id, the private message* and boolean value "*True*" for relevant and "*False*" for irrelevant. In the end, a manual inspection was done to see if any of the messages were labeled wrongly. An example is shown in Figure 4.1

4.5 Binary classification

For the binary classification task, we used a well known classifier Support Vecotor machine (SVM). A brief explanation of the SVM-classifier can be found in chapter 2.

A program to train the classifier was written using the scikit-learn library in python [34]. In the first step, the training dataset of 4000 messages, constructed in the previous step, was split into training and testing in order to train and measure the accuracy of the classifier. The data splitting was done using $sklearn.model\_selection.train\_test\_split()$ function, as it can read the input as arrays or matrices in order to create random shuffled training and testing sets. The ratio of train-test-split was set to 0.75 - 0.25.

To provide input to the classifier, the data was read using the python library pandas [33] which reads the csv-file constructed before. Further the column msg_post are vectorized using $TF-IDF$ Vectorizer from python, and represent the $x\_training$ and $x\_testing$ variables. While the column relevant is translated into numerical values 1 and 0 if msg_post is relevant or irrelevant respectively, and used as $y\_training$ and $y\_testing$ values.

When applying TF-IDF it is possible to save the features extracted as a vocabulary for further use. As can be seen in listing 4.1. Here we extracted 10,000 features for later use, and decided that each feature is one word in the document, where each private message is considered a document. The parameter analyzer informs TF-IDF vectorizer if n-grams of character should be used or as we did, called only word, meaning each word will be treated as a feature, but limited to 10 000. Strip accents makes sure that characters not representing Unicode characters are removed. then train and test vectors was "vectorized", based on the conditions giving from TF-IDF vectorizer. finally the vocabulary was saved.

Code listing 4.1: Convert text to vectorized features

```
#TF-IDF in python
vectorizer = TfidfVectorizer(strip_accents = 'unicode',
analyzer='word',max_features=10000)
tf_transformer = vectorizer.fit(toTrainX)
train_vectors = vectorizer.fit_transform(toTrainX)
test_vectors = vectorizer.transform(toTestX)
pickle.dump(tf_transformer, open("tfidf1.pkl", "wb")) #<-Saving vocabulary
```

Then the SVM classifier was trained on as can be seen in 4.2 before the trained classifier was saved by using the joblib library.

Code listing 4.2: SVM train and save

```
#SVM
from sklearn.svm import SVC
classifier = SVC(kernel='rbf', random_state = 42)
classifier.fit(train_vectors,y_train)
Y_pred = classifier.predict(test_vectors)
save  = joblib.dump(classifier,'classifiermodel.pkl')
```

and then loaded into a new script, where classifying the raw data without any labels was executed. When the classification process was completed, a new csv-file was created, containing all the relevant and irrelevant messages along with the all the private messages and the unique userID. The messages categorised as relevant, was extracted from the dataset predicted by the classifier, and saved for further analyses, while the irrelevant messages was discarded.

4.6 Multi-label classification

4.6.1 Manual labelling for training dataset

Creating labels to solve the multi-class problem is a hard task to accomplish. As the criteria for certain messages will suggest that a message can be relevant for multiple topics, hence a multi-label classification problem. In order to train a multi-label classifier, a training-set had to be created. This part was done manually, as there can be multiple correct labels. 508 messages was labeled, and saved as the training-set for the multi-label classification step. In the process of manually labeling, the decision of how many labels to create and what they should contain, was a tricky part. As the information we are mostly interested in, are the topics related to security and communications. There are usually a lot of other topics discussed in a hacker forum. By manually inspecting some of the private messages, a better understanding of how users talk to each other was achieved, and many topics are not directly involved with hacking and cyber security issues, such as, video and computer games. However, topics related to gaming and hacking are very often closely related. This work by manually reading some of the messages, is considered important, as a human usually have a better interpretation of what labels a message could contain.

Initially, we created 8 different labels, much like the approach in [28] also described in chapter 3. We used the following labels; "offering", "requesting", "minor hacking related", "advanced hacking related", "social", "gaming", "communication", "forum related". The label offering, was initially named selling, as many users sell accounts they have illegally obtained. However, when manually labeling the training-set, it was decided to include the all messages as would provide; some sort of help, sending configuration scripts and so. Thus "offering" is far more appropriate label than "selling". The label "Requesting" was first named buying, but after discovering that people not only wanted to buy stuff, but also asked for help, and requested access to topics etc. This label was renamed to "requesting" such as in [28]. "Minor hacking" includes all the users who are using security related words such as scraping, cracking, the tools used for this purpose etc. The label "Advanced hacking related" means the users who actually can provide some more insights, such as guiding users on how to set up configurations for different tools, explain how to spread malware, keyloggers and bypass antivirus. This category was pretty rare, yet a few messages was considered related to this label in the training-set of 500 messages. A message would be categorised as social, if it in-

cluded non security related words,neither did fit in the other categories, or used names of social media platforms.

To Gaming, all messages related to gaming, was labeled with this topic. Communication would be all messages, including the words related to communication; such as "add me on skype", "talk to you on" etc. Forum related, are the posts where users are complaining about other users on the forum, asks how to buy VIP-access or ask about topics and how to gain access. After creating the labels we created a heatmap to better understand the correlation between the features of the labels of the training set.

4.6.2 Training Multi-label classifier

In chapter 2 a brief outline of three different methods of solving a multi-label classification problems are discussed. As there exist many different algorithms and approaches, we went to test out a few of them, but only to proceed with one as was simple to implement but still provided good results. The three multi-label classification models as we tested are all based on Logistic Regression and belongs to the problem transformation category. The multi-label classifier we continued to proceed with was the classifier chains model [35], as we got good results and it was easy to implement.

Similar to binary classification where text needed to be converted to numerical values, we proceeded with the TF-IDF features for multi-label classifier too , as it is fast and easy to implement.

The classifier chains model takes each label and treat it as a binary classification problem as shown in listing 4.3. This approach is iteratively repeated for each label and predict the desired label for each unknown sample[2]. To accomplish this task we used the logistic regression library from [34].

[2] `https://www.kaggle.com/rhodiumbeng/classifying-multi-label-comments-0-9741-lb`,The original code can be found here, while in these thesis, it has been adapted and changed to fit the purpose of this thesis.

Code listing 4.3: Classifier chains

```
def feature(X, feature_to_add):
    '''
    Returns sparse feature matrix with added feature.
    feature_to_add can also be a list of features.
    '''
    from scipy.sparse import csr_matrix, hstack
    return hstack([X, csr_matrix(feature_to_add).T], 'csr')

for label in cols_target:
    print('Results for label: {}'.format(label))
    y = train_df[label]
    # train the model using X_train & y
    logreg.fit(X_train,y)
    # compute the training accuracy
    y_pred_X = logreg.predict(X_train)

    # make predictions from test_X
    test_y = logreg.predict(test_X)
    submission_chains[label] = test_y
    # chain current label to X_dtm
    X_train = add_feature(X_train, y)
```

After predicting new labels to the multi-label dataset, a correlation heatmap between the labels was generated to get better understanding of which features stick out. As we can see gaming-buying and selling- gaming have a strong correlation.

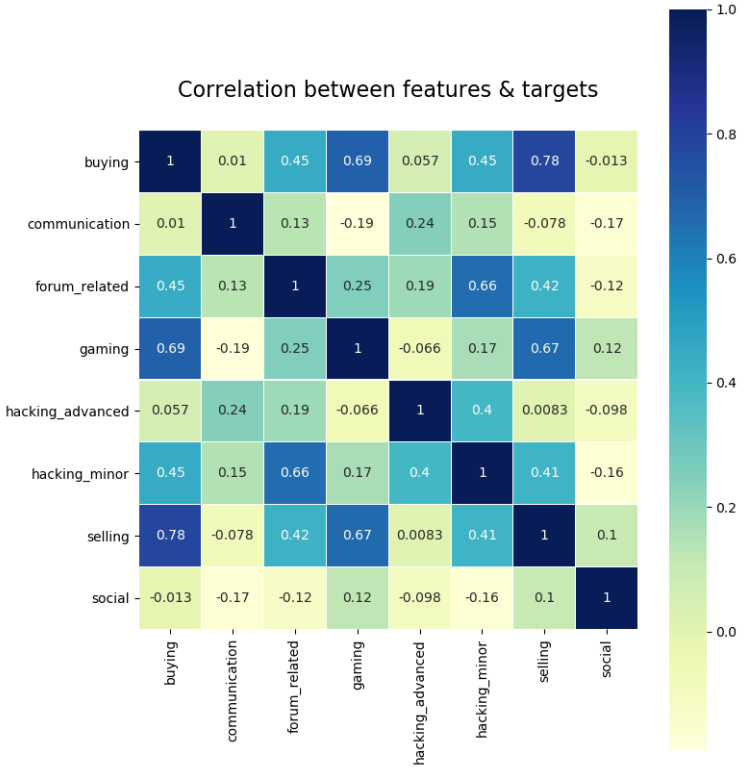


Figure 4.4: Correlation between features and target labels

Table 4.2: Vectorization

User-id	A1	A2	A3	A4	A5	A6	A7	A8
5	0	0	17	1	0	0	10	1
6	0	3	16	2	0	0	20	25
7	1	2	2	0	0	1	1	0
8	6	9	15	1	0	4	14	20
18	0	1	7	0	0	0	1	0

4.6.3 User Profiling: Creating Vectors of Size N

After the multi-labeling process, the new predicted dataset needed to be turned into vectors which could characterize users on hack forum and later used to perform clustering. As we now had a CSV file of 49932 rows, representing 9156 unique usernames, we would add each of the users messages in each respective category to create vectors. An example of how these vectors are created can be found in 4.2 where each entry of first column represent an individual and the corresponding row is created by adding up all the messages from that individual. Where A1-A8 represent the following;

A1 = offer; A2 = request; A3 = minor hacking activity; A4 = Advanced hacking activity; a5 = Social activity; A6 =gaming; A7 = communication; A8 = Forum related;

This was all done by a simple python script to sum up the rows. As we had 9156 users, we set different thresholds of the minimum number of total messages a users should have sent, in order to be further clustered. However, the original clusters of the 9165 users can be seen in 4.1 the rest are left for Appendix A

4.7 Clustering

As clustering is an unsupervised machine learning technique, we do not need to provide any training data, however some other parameters need to be provided in advanced. As a final step for the experimental part, at least for programming, we used the K-means algorithm briefly explained inchapter 2. the parameters needed to be set are the number of clusters and the maximum number of iterations the algorithm should run. As there are different techniques to decide the numbers of clusters, we went to use the well known "elbow-method" in order to decide. Using elbow method to decide the number of clusters, using K-means to create clusters. After the vectorization process, we had 9156 unique vectors (users) and applied K-means clustering in order to see if any useful results could be provided. The elbow method suggested three different clusters as can be seen in. Figure 4.5 We set a maximum number of 1000 iterations and random state = 42, using the K-means clustering algorithm provided by Scikit- learn library [34].

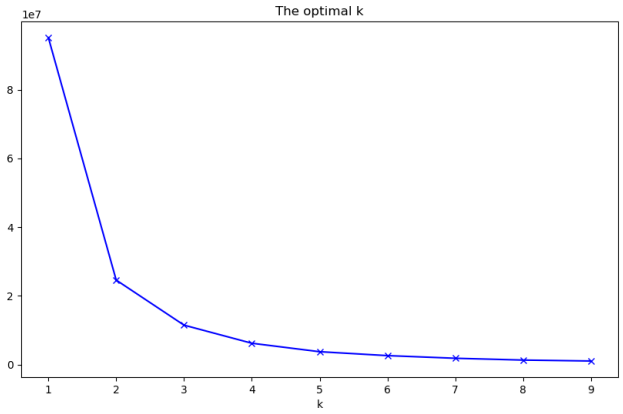


Figure 4.5: Elbow method

Table 4.3: Users represented by vectors

User_ID	selling	buying	Hacking minor	Hacking advanced	social	gaming	communication	forum_related	Total labled messages	Total messages	Group
442288	67	118	156	1	0	49	15	25	431	1363	3
2902	2	13	18	1	0	2	70	9	115	738	15
15398	1	15	103	2	0	0	52	16	189	2287	44
13521	8	25	95	3	0	8	19	17	175	906	12
142983	5	2	0	0	0	1	48	0	56	298	15
208840	34	86	46	0	0	53	8	19	246	1099	3
51349	50	73	145	1	0	37	24	22	352	5593	12
124673	2	3	5	0	0	0	38	3	51	585	10
34893	14	21	39	1	0	30	42	12	159	1861	10
61078	16	29	69	2	0	10	39	14	179	2973	12

4.7.1 Validation

After the whole programming process was completed we were left with a cluster who points out potential interesting users. To verify if the model failed or not, manual inspection on some of the users was done. As the original dataset was stripped down, to only be readable for a machine, we went back to extract the private messages of these users to read their content. First we only extracted the vectors created in the vectorization step in order to understand which labels scored most points. We choose to check the user found in Table 4.3, as they from a clustering analysis perspective sticks out. Inspecting more than 10 users, would also exceed the scope of this thesis. Messages extracted from the selected users can be seen in Table 4.5.

Table 4.4: Group Structure of Nulled.io

Group Name	Members	Administrators	Moderators	VIP	Reverser	Donator	Contributor	Royal
Number	3	4	6	7	8	9	10	12
Group Name	VIP	Legendary	Senior Moderator	Semi-admin	Legendary Reverser	VIP_Plus	Support	Administrators
Number	15	38	39	41	44	47	48	49

Table 4.5: Manual inspection of private messages

msg_author_id	msg_post
2902	hi bro can u help me run bol ? thx in advice my skype is giwrgos_nexus
2902	i crack it alone dude can u contact me on skype?i will give u
2902	tell me how mutch u want to teach me how to sql inject
2902	hey i see your website u seem like u know how to scam someone and get into his files can u contact me on skype rdca_83 i need u :D
2902	well at your website u have posted a botnet can i take some stuff from one guy with that? i want some files from his pc not passwords
13521	I've tried to used kevin bot I open cracked bol with VIP then kevin bot and it says an authenticator problem and I've did everything as the instructions. Can you please help me? ty
13521	I think I found it it lags the game and if you continue lagging it for 5 minutes it might erase the game Do you think it is detected? It's a server side hack I think
13521	Are you a scripter? I never worked with .lua before
13521	you're only disconecting your IP right? or the entire match? Can you drop only one enemy from the enemy team? thanks for the video btw
13521	did you got any lol account lvl 30 with that program?
13521	Create a nulled.io acc with the same name as a buyer from SAC:R log in with that account using bol vip loader. I use that to bypass every script I think evadeee is not workin tho.
15398	I can't add your Skype preffer to deal by pms i'm tracked by many ppl.
15398	I however need to know which game/program loads these models eva or any tool that can read them.
15398	ok add me to skype xbroly7
15398	it's a private versionm that only such a few ppl have
15398	i'm using IDA pro and Ollydbg and start from some basi tutorials youtube maybe
34893	Can i know why you ain't going to update your hack anymore
34893	Remember that time I asked if you were accepting crack requests?
34893	No I was trading a CS:GO account for a br account on league of legends but that was a long time ago
34893	No I don't have any experience with hack or something like that all I know is gfx
51349	can you give me a config for Sentry mba for league of legend na plz
51349	Wow i really wanted to say to you Good Job man those account are amazing but they are all valided i had no luck :/. Well just wanted to pm you to say that you are doing a good work
51349	my host file is completely empty i did search for over 2 hour but i dint found anything help me please
51349	you want more? im going to crack a lot of account tomorow i will post like 10 or more if you want
51349	are you using sql or you make combo whit a list of username to find account?
61078	I found a topic of a guy and he asked for someone to hack his website And you hacked it I have to say good job mate.
61078	Ohh..Btw how you crack league accounts bro?
61078	I want to find the ip and i will try to ddos it
61078	Can i give you my skype and talk from there? I hate messages on nulled :P
61078	I use metasploit kali linux virtualbox btc mining (sometimes) sentry mba skype dubrute some timesez crack :P
124673	Hey I followed your method but i have one problem I can't use the plugin Can you please help me? Thanks My skype to contact me at any time
124673	Scraped and applied custom password very big PW list :D
124673	How much are you willing to spend? you can get ultimate combolist for a region of your choise with 5 EUR
124673	I can give you one cheap :) Add me on skype for faster communication
124673	Hey I got what you need please add me on skype for faster transaction
142983	Had to fake two amazon receipts Said that the battery was not working and pretty much worked
142983	Im sorry but I cannot simply afford to waste my time on people who are not able to purchase what I am selling
142983	I dont accept PayPal. Only BTC. the answer is no
142983	Please send 0.19 BTC to
142983	Also let me know when you have sent them and add me on skype golltes
208840	Awesome. In case you sell them and i am not answering PM's here feel free to add me at my Skype tomynica. I am online there most of the day
208840	Hello i can help you if you want. I cracked verified accounts easily. Give me the username and password of that account and i will see what i can do. What server?
208840	I got 2700 combo/min most of the time. Got 5500 just for a few minutes. Using 210 bots is that too much?Btw any tips on how to crack better since you are a lot experienced?
208840	Anyways feel free to PM me here or at Skype once you start with that harder selling got like 10 accounts to sell including my main from season 1 that is not cracked obviously
208840	What kind of dorks mixed/gaming? Already leaked and checked?
442288	Hey do you want to teach me how to effectivly crack League of Legnds accounts and i can give you some nice Origin accounts with games that you want in return?:)
442288	That is strange i never had that issue. What is version of Sentry MBA that you are using?
442288	Same thing. Have you tried to crack PSN accounts? Does it works for you?
442288	Hello config is not for sale i am selling only Amazon accounts. Are you interested?
442288	How much for 10 CS:GO keys?

Chapter 5

Experimental setup and results

This chapter are providing information about the experimental setup and the results achieved in this thesis.

5.1 Setup

For the programming part we had the following hardware at our disposal. Please see the list below. The programming language used was python 3.6.2 64-bit. The code was written in the lightweight, but powerful source code editor Visual Studio Code, where several packages from Scikit-learn [34] and [33] were imported and used in the programming process.

- For the experimental part, a relatively power full desktop computer was used.
- CPU: AMD Ryzen 5 2600X 6 cores- 12 threads clock frequency up 4.25 Ghz
- Memory : 16 GB 2666 MHz
- GPU: Nvidia RTX2060- 6 GB video memory
- Storage: 500 GB M2. SSD

```
----------------------------------------------------
Accuracy of Naive Bayes on train:  0.9753333333333334
Accuracy of Naive Bayes on test:  0.875
----------------------------------------------------
Accuracy Of SVM on training set :  0.9966666666666667
Accuracy Of SVM on testing set  :  0.929
```

Figure 5.1: Accuracy of SVM

5.2 Binary classification

As we used SVM as the binary classifier the results of training accuracy can be shown in Figure 5.1 where we achieved an accuracy of 92.29% on the testing set.

5.3 Multi label classification

The results from the multi-label classifier: classifier chains can be seen in 5.2. As we can see we achieved pretty good accuracy on the training data, and F-score and hamming loss is very good. With this results at hand we felt confident in moving forward by creating vectors of the now multi-labeled dataset.

5.4 Clustering

As the final step, we are left with 9156 unique users, each of them are represented in an 8-dimensional vector. To see if any of the users are standing out, we used clustering. As there are 9156 users present, there are some difficulties in identifying all of the users. However, we are interested in those who are standing out, which is what we achieved if we look at Figure 5.3. Each number represents the unique user-Id for each user, making it easier to identify the users who are standing out. This was used as the decision to manually inspect some of the users.

5.4.1 Manual inspection

After clustering analysis was completed, we could look further into some of the users to be able to answer the research questions asked in chapter 1. In Table 4.5 extracted messages from the users used as a comparison in Table 4.3 are presented.

```
Results for label: selling
Training Accuracy is 0.997920997920998
Classifier chains  F1-score: 0.998
Classifier chains Hamming Loss: 0.002
Results for label: buying
Training Accuracy is 0.9958419958419958
Classifier chains  F1-score: 0.996
Classifier chains Hamming Loss: 0.004
Results for label: hacking minor
Training Accuracy is 1.0
Classifier chains  F1-score: 1.0
Classifier chains Hamming Loss: 0.0
Results for label: hacking advanced
Training Accuracy is 0.9958419958419958
Classifier chains  F1-score:
 0.996
Classifier chains Hamming Loss: 0.004
Results for label: social
Training Accuracy is 0.9958419958419958
Classifier chains  F1-score: 0.996
Classifier chains Hamming Loss: 0.004
Results for label: gaming
Training Accuracy is 0.9937629937629938
Classifier chains  F1-score: 0.994
Classifier chains Hamming Loss: 0.006
Results for label: communication
Training Accuracy is 1.0
Classifier chains  F1-score: 1.0
Classifier chains Hamming Loss: 0.0
Results for label: forum related
Training Accuracy is 0.997920997920998
Classifier chains  F1-score:
 0.998
Classifier chains Hamming Loss: 0.002
```

Figure 5.2: Results Multi-labeling

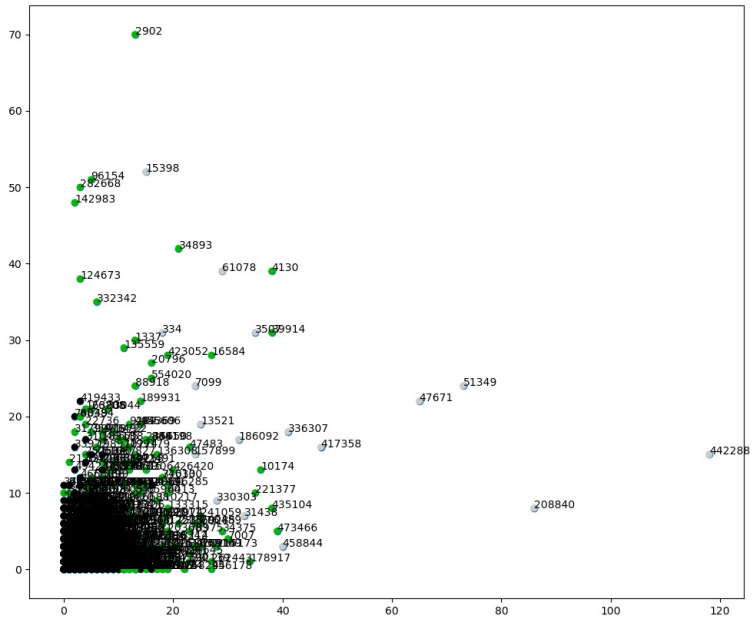


Figure 5.3: Result of clustering 9156 users

	Users	Private messages
Original	599065	800539
New dataset	9156	49932
percent of filtered users& messages	1,52%	6,23%

Table 5.1: Reduction of users based on methodology

5.4.2 Results

If we look at the initial goal, "Develop a model that can highly benefit the forensics investigation process" We argue that this thesis has contributed to this task. By looking at the methodology block diagram (Figure 4.2), we can say that this approach can be integrated as a part of the Digital forensics investigation process. To prove this statement, we can use the results from this model to answer the research questions. First of all, when we extracted all the private messages from the database, We created a dataset with 800539 private messages(including duplicates) written by 599065 unique users. After following the approach proposed in Figure 4.2, we are left with only 9156 users and 49932 private messages, which are about only 1.5% of the users and only 6% of the private messages in the hacker forum. The results can be seen in Table 5.1 An in-depth discussion related to the research questions asked in chapter 1 can be found in chapter 6.

Chapter 6

Discussion & conclusion

In this chapter, we discuss different topics related to the task accomplished and discuss how the research questions have been answered, before we finally conclude.

6.1 Research question

The following three subsections will discuss the findings related to the proposed research questions.

6.1.1 Research question 1

1. How to create a well defined criterion, to separate relevant posts from irrelevant posts?
2. How to use the well-defined criterion to distinguish between Key-actors and Script kiddies?

A well-defined criterion for separating messages was already established in [26] before we started this project. However, we adopted the approach and applied it to the private messages of the nulled.io hacker forum. To separate script kiddies from key-actors, we first created multiple labels to distinguish messages from each other, followed by creating n-dimensional vectors from these labels for each user. Finally, we used clustering algorithm to discover some of the users stick out. After the clustering process, we conducted a manual inspection of the users to see what they are talking about (see Table 4.5). Unfortunately, it is not clear from the messages that we have been able to locate any key-actors, but the approach seems to be sound, as out of the users we picked only two of them to belong to the group "members-group: 3" when we compare Table 4.3 and Table 4.4. While the rest belongs to the upper layer of the hierarchy of the nulled.io hacker forum.

6.1.2 Research question 2

1. How to identify different channels/categories of communication?

2. When do actors decide to use another channel to communicate?
3. Why do potential users of interest decide to change the channel of communication?

Identifying different channels of communication, was a part of the process all along. From the manual inspection analysis including the manual labeling part, it seems like Skype is the primary channel of communication outside of nulled.io. While in past time IRC was pointed out as the primary channel, hackers used for communication [27]. Out of the 10 users we choose to inspect manually, 6 of them mention Skype at least once. However, one user mention Skype in the context; "I can't add your skype preffer to deal by pm's i'm tracked by many ppl."- user 15398. It is unknown if this claim is true or not, as the same user later on in a conversation says; "Ok add me to Skype". It is a possibility that the user wants to act, as he is a bigger deal than he is, a quick google search on his username, provides his twitter profile, and certainly does not appear, to be a hot-shot from a hacker community, as a hacker of interest would probably keep a lower profile, but this would be speculation at best. For the next two questions, "When do actors decide to use another channel to communicate?" and "Why do potential users of interest decide to change the channel of communication?" It seems we haven't dug deep enough, as this could be carried out a separate project. There are however some indications from the messages provided in Table 4.5 which indicates that users that choose to use other channels such as Skype, mainly do so, when they need help or want to sell or buy something.

6.1.3 Research question 3

- How can a list of possible attributes/features be extracted from the content of a hacker forum, be used to profile a malicious actor?

To answer the last research question, " How can a list of possible attributes/features be extracted from the content of a hacker forum, be used to profile a malicious actor?" We believe that it is possible to profile malicious actors based on our approach. By first filter out irrelevant messages and users, followed by creating multiple labels for each users, to further create N-vectors based on the information we are after. These N-vectors will then create the basis for further analysis, we choose to use clustering, followed by manual inspection. But other methods could also be applied, for example use the features created in combination of the Genetic Algorithm proposed in [7].

6.1.4 Summary of research questions

As we have shown, we have created a method for mining actors from hacker forum, based on certain criteria, further, we have partially identified channels of communication, and we have demonstrated one approach on how to extract features of a hacker forum. To compare our results from related work, we would like

Private centrality results									
UID	C_{deg-}	UID	C_{deg+}	UID	C_B	UID	C_C	UID	C_E
1	0.08412	1	0.42331	1	0.41719	1	0.40665	61078	0.45740
1471	0.05028	51349	0.00773	1471	0.02369	51349	0.28442	51349	0.30353
1337	0.04289	88918	0.00695	334	0.02286	88384	0.28102	1	0.24505
8	0.03970	47671	0.00617	1337	0.02253	10019	0.28080	88918	0.21214
15398	0.03967	334	0.00600	15398	0.02129	61078	0.28043	193974	0.19651

Figure 6.1: Individuals identified by centrality measures

to draw a comparison to [6]. By using the Figure 6.1 provided from the result section of [6]. Where actors were identified by using centrality measures. We can see that our approach point out the same user -15398 Figure 5.3 as a potential actor of interest. However, we would like to address the issue here, where we identify this user because the user scored high in certain topics such as minor hacking and communication, not just because of being an active member of the hacker forum nulled.io. The other users in Figure 6.1 are not identified in our cluster unless we set a threshold for a minimum number of messages, the drawback of setting such a threshold is the users we potentially are going to miss, as we believe the number of messages is not the only criteria for being an interesting actor of a hacker forum.

6.2 Discussed topics

In this section, we discuss topics related to the master thesis and some important areas when the end goal is to identify users in online forums, by applying machine learning techniques.

6.2.1 Private messages vs public posts

In this thesis, we explored the private messages between users of a hacker forum. While a lot of research focuses on the public available forum posts. When we have access to the whole database of a hacker forum, we thought this would be a unique opportunity to explore the insight of a hacker forum. The potential findings in private messages can be huge, as even for a hacker forum who have hidden sections only available for certain users. There will always more be more shady content in private conversations, as users won't expose themself necessary in the public forums. When the whole dataset was available after all, and the public forum post to a certain degree was already experimented with, we decided the private conversations would be more interesting to research on.

6.2.2 Privacy and ethical aspects

Users on the hacker forum might have the skill-set to appear anonymous online. Thus the need to investigate and analyze such a forum is present. However, a lot of users might only sign up to learn based on their interest and passion, and do

not aim to commit any crimes with the knowledge they might get from signing up on a hacker forum. Is it then fair that we potentially develop methods that put a misleading target on their backs? We can however justify with some certainty that the research is necessary, as supervised machine learning algorithms can appear to be a black box we never fully understand, we can't leave hacker forums to be a black box. As most of the users of such forums probably never commit serious crimes, some still use their gained knowledge to commit illegal activities. But as long as there exist active research in this area, it might make some users think twice before they cross a line. Hunting for people, to identify criminals especially when using machine learning, needs to be discussed. As we can not always predict why certain machine learning algorithms predict a certain output. And this in particular is true for unsupervised machine learning algorithms. There is always a margin of error, which might let some interesting users slip away, while some "unlucky" users of such forums, will be in the loop. Putting targets on the backs of allegedly innocent users is a thought need to be kept in mind while actively investigating such forums. The development of methodology as we have done here can also be misused, and therefore needs to be dealt with caution. after all, everyone is innocent until proven guilty.

6.2.3 Suggestions for improvement

As the development of this model is based on different steps of text classification by using machine learning algorithms. Creating better input features to fit the machine learning components in 4.3 can go a long way to identify Key-actors. The key is to make the first training set as good as possible, however, this is very resource-demanding as it requires manual work. The same goes for creating good labels for multi-label classification. But it will generate a better basis for extracting the relevant features in the end. We believe less general labels, and more specific labels for solving the multi-label classification part, will be of an advantage when the goal is to profile malicious actors. Also apply the use of word-embeddings as a part of feature extraction instead of a statistical method as TF-IDF are, can be helpful. Such an approach would capture the actual context of the messages and can improve the model even further. However, it is harder to explain how a two-layer neural network works, such as word2vec, in court, rather than TF-IDF, if handling a criminal case. It could also be useful to split the parts, related to communication and security-related information, as the scope turns quite large when handling both issues at the same time.

Stopwords pros cons

When handling text classification problems, almost every tutorial and guide out there explains the importance of removing stopwords. And in our case, it was the right thing to do, as we used a statistical approach to create numerical features of the text, TF-IDF. However, if taken this a step further and using word embedding created by models such as word2vec [17]. It is important to remember that the

meaning of sentences, can get lost if the stop words are removed[36]. There is a trade-off in terms of speed and simplicity when exploring models such as we have done in this thesis. And even we have explored and explain e.g word2wec, we did not use it to create word embeddings. As for this project, we should prove that our methodology works and the need to make it more complicated than necessary was seen as a trade-off, in favor of TF-IDF, and thus the removal of stopwords is necessary.

6.2.4 Strange findings and discoveries

Initially, we provided the results based on all the 9156 users who filtered as relevant. As we can see in FigureFigure 5.3 providing a cluster of 9156 users, is not necessarily very informative. However, a few users stand out and allow us to manually inspect why they are a deviation from the rest of the group. However, we experimented further by filtering on a threshold of minimum 10,20,30, 50, and 70 messages Appendix A. But as this indicates we are rewarding the users who are actively sending private messages. Much like in [6], with one difference and that's all the users that are extracted, are active with relation to certain topics. We would like to point at that by setting the threshold of minimum 50 messages, the number of users is reduced to only 145 which is significantly lower than the original number of users, thus it is easier to investigate those users for criminal activity. However, being active on a hacker forum, by sending messages, is not automatically an indication that you are doing something criminal. When filtering this way, some users who might send fewer private messages, but rather send very relevant messages, will be lost. The ideal approach would be to go back and improve the labels, both for the binary classification task and the multi-label classification step.

6.2.5 Filtering on threshold of posts

Initially, we provided the results based on all the 9156 users who filtered as relevant. As we can see in FigureFigure 5.3 providing a cluster of 9156 users, is not necessarily very informative. However, a few users stand out and allow us to manually inspect why they are a deviation from the rest of the group. However, we experimented further by filtering on a threshold of minimum 10,20,30, 50, and 70 messages Appendix A. But as this indicates we are rewarding the users who are actively sending private messages. Much like in [6], with one difference and that's all the users that are extracted, are active with relation to certain topics. We would like to point at that by setting the threshold of minimum 50 messages, the number of users is reduced to only 145 which is significantly lower than the original number of users, thus it is easier to investigate those users for criminal activity. However, being active on a hacker forum, by sending messages, is not automatically an indication that you are doing something criminal. When filtering this way, some users who might send fewer private messages, but rather send very relevant messages, will be lost. The ideal approach would be to go back and improve the labels, both for the binary classification task and the multi-label classification step.

6.2.6 Conclusion

The development of a model that can benefit the digital forensics investigation process, has been finalized. The model is nowhere near perfect, as it is built on a proof of concept. However, it shows that it is possible to narrow down the search for potential interesting users on a hacker forum. As the results show we could reduce the pool of users by 98.5%. And we can argue that the model in Figure 4.3, can fit into the Digital investigation process as it can be seen as an iterative process, but with room for improvement. It will fit well into the sections of the examination and analysis phase of collecting evidence. We believe that the model we have developed in this thesis would be of potential use, in a forensics investigation case, where an online forum is under investigation. It is also worth to notice, that chaining the features to adapt to a different domain, is relatively trivial, but time-consuming. We have been able to answer all the research questions, with some variation of success. We consider the first category of research questions are answer appropriately. In the second category related to communication, we can we some certainty that the users of nulled.io turn to Skype as the channel of communication, when not using forum private messages. On the other hand, create an exact statement of when the users, change their channel of communication is proven difficult to predict. The same goes for why users change the channel of communication, as it seems the users mostly spent time asking for help, providing help, or selling each other illegally obtained accounts, before turning to Skype.

Chapter 7

Further work

As this projects have only briefly touched upon how to apply different machine learning techniques on a large dataset in terms of mining hackers, it means there is much left to explore. The models and techniques used, have produced satisfying results for us at this stage, but exploring the same methodology but with different algorithms is an option. We believe capturing more of the context in both private communication and public forum posts, in hacker forums, is something that needs to be explored further. Our final step in the model we have developed is using clustering, but it is not sure that this is the ultimate approach. There are probably other ways of benefit of the vectors created. Another topic is the questions asked about communication, as we couldn't find very much relevant research in this area, this is something that definitely should be researched further.

Bibliography

[1] M. Kan, 'Newegg hacked to steal customers' credit card data', Sep. 2018. DOI: https://uk.pcmag.com/news-analysis/117521/newegg-hacked-to-steal-customers-credit-card-data.

[2] S. Pastrana, D. Thomas, A. Hutchings and R. Clayton, 'Crimebb: Enabling cybercrime research on underground forums at scale', Apr. 2018, pp. 1845–1854, ISBN: 978-1-4503-5639-8. DOI: 10.1145/3178876.3186178.

[3] M. STAFF, 'The motherboard e-glossary of cyber terms and hacking lingo-hacking from a-z', Jul. 2016. [Online]. Available: https://www.vice.com/en_us/article/mg79v4/hacking-glossary.

[4] R. Security, 'Nulled.io: Should've expected the unexpected!', May 2016. DOI: https://www.riskbasedsecurity.com/2016/05/10/nulled-io-shouldve-expected-the-unexpected/.

[5] Z. Fang, X. Zhao, Q. Wei, G. Chen, Y. Zhang, C. Xing, W. Li and H. Chen, 'Exploring key hackers and cybersecurity threats in chinese hacker communities', in *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, Sep. 2016, pp. 13–18. DOI: 10.1109/ISI.2016.7745436.

[6] J. W. Johnsen and K. Franke, 'Identifying central individuals in organised criminal groups and underground marketplaces', in *Computational Science – ICCS 2018*, Y. Shi, H. Fu, Y. Tian, V. V. Krzhizhanovskaya, M. H. Lees, J. Dongarra and P. M. A. Sloot, Eds., Cham: Springer International Publishing, 2018, pp. 379–386, ISBN: 978-3-319-93713-7.

[7] E. Marin, J. Shakarian and P. Shakarian, 'Mining key-hackers on darkweb forums', in *2018 1st International Conference on Data Intelligence and Security (ICDIS)*, Apr. 2018, pp. 73–80. DOI: 10.1109/ICDIS.2018.00018.

[8] J. Waldman and E. Cordona, 'Top 25 threat actors – 2019 editionby sbs cybersecurity, llc', Dec. 2019. [Online]. Available: https://sbscyber.com/resources/top-25-threat-actors-2019-edition.

[9] C. H. News, 'Top 10 most notorious hacking groups of all time', Jul. 2016. [Online]. Available: https://cyware.com/news/top-10-most-notorious-hacking-groups-of-all-time-32d01ba2.

[10] G. Palmer, 'A road map for digital forensic research," technical report (dtr-t001-01) for digital forensic research workshop (dfrws),new york,', Jul. 2001. [Online]. Available: https://dfrws.org/sites/default/files/session-files/a_road_map_for_digital_forensic_research.pdf.

[11] 'The digital forensics process', in *Digital Forensics*. John Wiley Sons, Ltd, 2017, ch. 2, pp. 13–49, ISBN: 9781119262442. DOI: 10.1002/9781119262442.ch2. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781119262442.ch2. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119262442.ch2.

[12] I. Kononenko and M. Kukar, *Machine Learning and Data Mining: Introduction to Principles and Algorithms*. Horwood Publishing Limited, 2007, ISBN: 1904275214.

[13] J. Huang, G. Li, S. Wang, W. Zhang and Q. Huang, 'Group sensitive classifier chains for multi-label classification', in *2015 IEEE International Conference on Multimedia and Expo (ICME)*, 2015, pp. 1–6.

[14] P. C.P, K. Jyothy and B. Noora, 'Multi label classification based on logistic regression (mlc-lr)', Sep. 2016, pp. 2708–2712. DOI: 10.1109/ICACCI.2016.7732470.

[15] S. Decherchi, S. Tacconi, J. Redi, A. Leoncini, F. Sangiacomo and R. Zunino, 'Text clustering for digital forensics analysis', vol. 63, Jan. 2009, pp. 29–36. DOI: 10.1007/978-3-642-04091-7_4.

[16] K. S. Jones, 'A statistical interpretation of term specificity and its application in retrieval', *Journal of Documentation*, vol. 28, pp. 11–21, 1972.

[17] X. Rong, 'Word2vec parameter learning explained', 2014. arXiv: 1411.2738 [cs.CL].

[18] T. Mikolov, G. Corrado, K. Chen and J. Dean, 'Efficient estimation of word representations in vector space', Jan. 2013, pp. 1–12.

[19] J. Ami-Narh and P. Williams, 'Digital forensics and the legal system: A dilemma of our times', *Australian Digital Forensics Conference*, Jan. 2008.

[20] S. T.-r. by Threathunting.net, 'The threat hunting reference model part 2: The hunting loop', Oct. 2015. [Online]. Available: https://www.threathunting.net/files/The%20Threat%20Hunting%20Reference%20Model%20Part%202_%20The%20Hunting%20Loop%20_%20Sqrrl.pdf.

[21] R. Puzis, P. Zilberman and Y. Elovici, *Athafi: Agile threat hunting and forensic investigation*, 2020. arXiv: 2003.03663 [cs.CR].

[22] K. Wafula and Y. Wang, 'Carve: A scientific method-based threat hunting hypothesis development model', in *2019 IEEE International Conference on Electro Information Technology (EIT)*, 2019, pp. 1–6.

[23] O. Elezaj, S. Y. Yayilgan, E. Kalemi, L. Wendelberg, M. Abomhara and J. Ahmed, 'Towards designing a knowledge graph-based framework for investigating and preventing crime on online social networks', in *E-Democracy – Safeguarding Democracy and Human Rights in the Digital Age*, S. Katsikas and V. Zorkadis, Eds., Cham: Springer International Publishing, 2020, pp. 181–195, ISBN: 978-3-030-37545-4.

[24] M. Nouh, J. Nurse and M. Goldsmith, 'Towards designing a multipurpose cybercrime intelligence framework', Sep. 2016. DOI: 10.1109/EISIC.2016.018.

[25] E. Nunes, A. Diab, A. Gunn, E. Marin, V. Mishra, V. Paliath, J. Robertson, J. Shakarian, A. Thart and P. Shakarian, 'Darknet and deepnet mining for proactive cybersecurity threat intelligence', in *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, Sep. 2016, pp. 7–12. DOI: 10.1109/ISI.2016.7745435.

[26] I. Deliu, C. Leichter and K. Franke, 'Extracting cyber threat intelligence from hacker forums: Support vector machines versus convolutional neural networks', in *2017 IEEE International Conference on Big Data (Big Data)*, Dec. 2017, pp. 3648–3656. DOI: 10.1109/BigData.2017.8258359.

[27] V. Benjamin, W. Li, T. Holt and H. Chen, 'Exploring threats and vulnerabilities in hacker web: Forums, irc and carding shops', in *2015 IEEE International Conference on Intelligence and Security Informatics (ISI)*, 2015, pp. 85–90.

[28] A. Caines, S. Pastrana, A. Hutchings and P. J. Buttery, 'Automatically identifying the function and intent of posts in underground forums', 1, vol. 7, Nov. 2018, p. 19. DOI: 10.1186/s40163-018-0094-4. [Online]. Available: https://doi.org/10.1186/s40163-018-0094-4.

[29] S. Müller and F. Ulrich, 'The competing values of hackers: The culture profile that spawned the computer revolution', vol. 2015, Mar. 2015, pp. 3434–3443. DOI: 10.1109/HICSS.2015.413.

[30] T. W. Edgar and D. O. Manz, 'Chapter 3 - starting your research', in *Research Methods for Cyber Security*, T. W. Edgar and D. O. Manz, Eds., Syngress, 2017, pp. 63–92, ISBN: 978-0-12-805349-2. DOI: https://doi.org/10.1016/B978-0-12-805349-2.00003-0. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780128053492000030.

[31] T. W. Edgar and D. O. Manz, 'Chapter 5 - descriptive study', in *Research Methods for Cyber Security*, T. W. Edgar and D. O. Manz, Eds., Syngress, 2017, pp. 131–151, ISBN: 978-0-12-805349-2. DOI: https://doi.org/10.1016/B978-0-12-805349-2.00005-4. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780128053492000054.

[32] P. D. L. J. E. Ormrod, 'Practical research; planning and design', in *Practical Research; Planning and Design*, T. W. Edgar and D. O. Manz, Eds., Pearson, 2015, pp. 154–193, ISBN: 978-1-292-09587-5.

[33] T. pandas development team, *Pandas-dev/pandas: Pandas*, version latest, Feb. 2020. DOI: 10.5281/zenodo.3509134. [Online]. Available: https://doi.org/10.5281/zenodo.3509134.

[34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, 'Scikit-learn: Machine learning in Python', *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[35] J. Read, B. Pfahringer, G. Holmes and E. Frank, 'Classifier chains for multi-label classification', in *Machine Learning and Knowledge Discovery in Databases*, W. Buntine, M. Grobelnik, D. Mladenić and J. Shawe-Taylor, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 254–269, ISBN: 978-3-642-04174-7.

[36] L. Wikarsa, *Does pre-processing step (remove stop word) effect sentiment analysis result?*, Sep. 2018.

Appendix A

Additional Material

Code listing A.1: Extracting code from Sql database

```
import pandas as pd
import mysql.connector
import csv


connection = mysql.connector.connect(host ="localhost",
user = "root", password='*********', database = "nulled")

print(connection,"Connetion established")

databaseCursor = connection.cursor()
query1 = pd.read_sql_query('''SELECT member_id,
member_group_id from members where member_id =495''',connection)
query2 = pd.read_sql_query('''SELECT msg_author_id, msg_post
From message_posts WHERE msg_author_id = 495''',connection)
#query3 = pd.read_sql_query('''SELECT * From topics ''',connection)
#query4 = pd.read_sql_query('''SELECT * From message_topics ''',connection)
WHERE member_group_id = 49  ''',connection)
#query = pd.read_sql_query('''SELECT  pid, author_id, author_name,
#post FROM posts WHERE author_id = 53228 IN
#(SELECT member_id FROM members WHERE member_group_id = 8 ) ''',connection)

framesToSelect = query1
framesToSelect1 = query2
#framesToSelect = query2
print(framesToSelect1)
print(framesToSelect)
print('Query done')
```

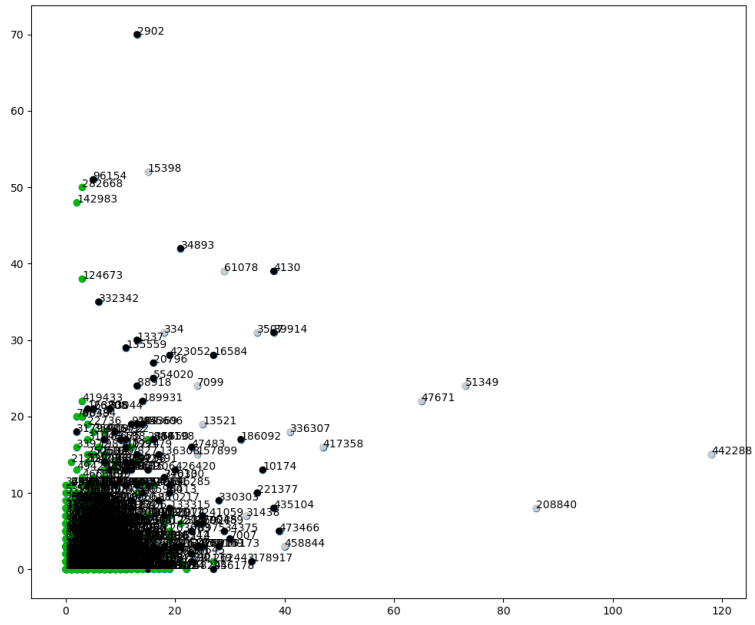


Figure A.1: Filtering on minimum 10 messages = 1113 users

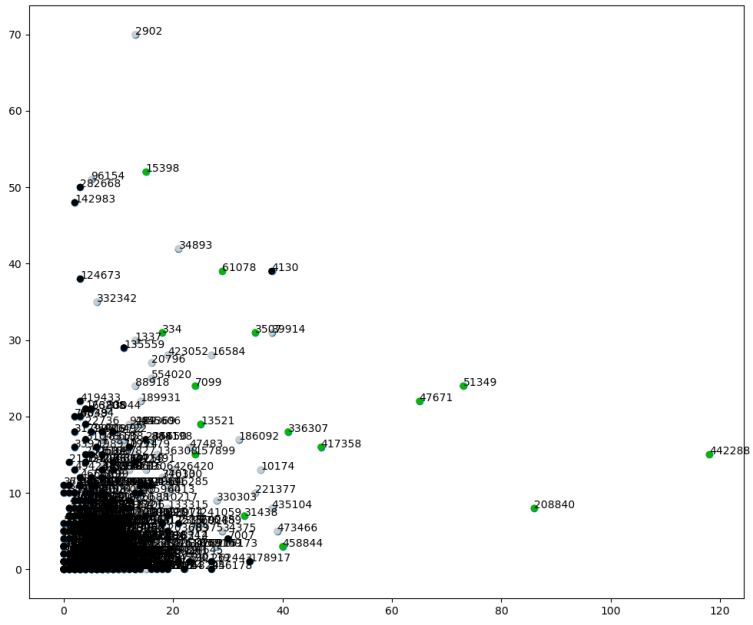


Figure A.2: Filtering on minimum 20 messages = 516 users

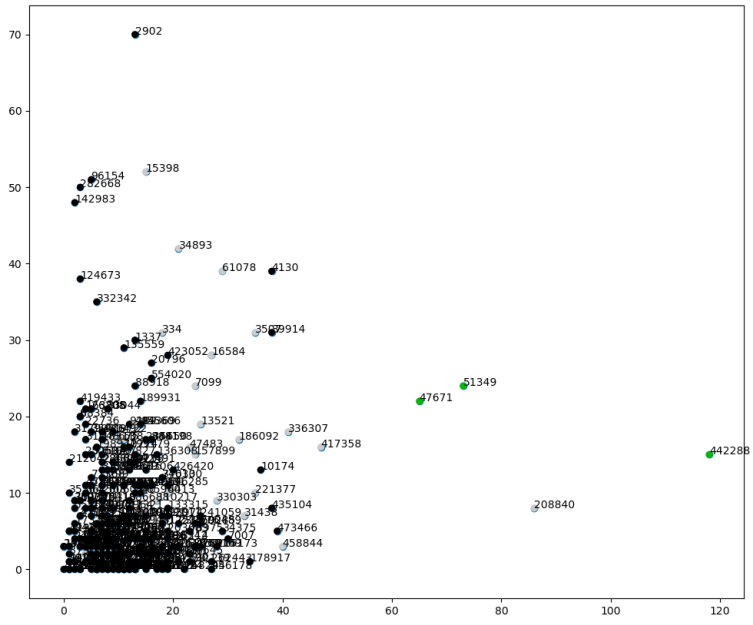


Figure A.3: Filtering on minimum 30 messages = 300 users

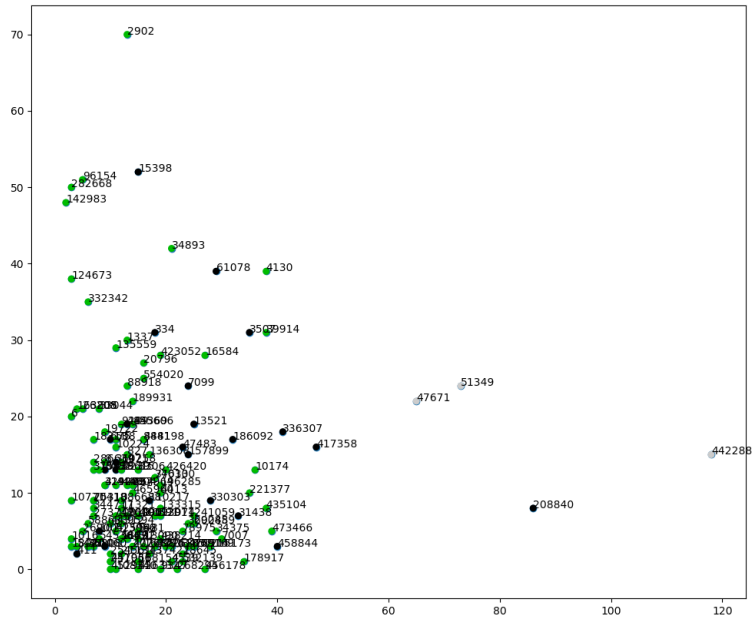


Figure A.4: Filtering on minimum 50 messages = 145 users

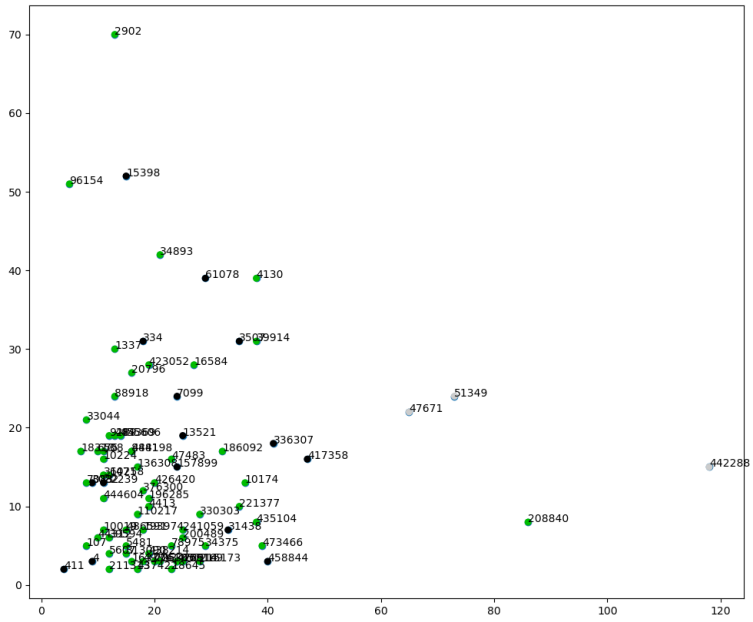


Figure A.5: Filtering on minimum 70 messages = 81 users