University of Macedonia

School of Information Sciences

Department of Applied Informatics

BACHELOR THESIS

Extending the Cybercrime Incident Architecture with a feature-based Cybercrime Classification System (CCS)

Dimitrios Vavatsioulas

Supervisors

Sophia Petridou

Kostas Vergidis

Thessaloniki, Greece

June 2021

Acknowledgements

Throughout the process of writing this Bachelor thesis, I have received tremendous help and support from many people.

I would first like to thank my supervisors, Professors Sophia Petridou and Kostas Vergidis, whose expertise was invaluable in formulating the research methodology. Your informative feedback and guidance assisted me to sharpen my thinking and raise the quality of my work. Our collaboration will always be an inspiration for me and I feel grateful that I had the chance to work with you and carefully observe your exceptional way of thinking.

I would also like to thank my professors, Georgios Evangelidis and Georgia Koloniari, for the valuable knowledge that I have gained from you throughout my studies and inspired me to combine Data Mining with the Cybercrime domain. I am also thankful to the Applied Informatics department and all its member's staff for all the considerate guidance I have received.

Finally, I would like to thank my family and my parents for their wise counsel and for their continuous support during my studies. I would not have been able to be so devoted and focused on my studies if you were not by my side for these 4 years. In addition, I could not have completed this thesis without the support of my close friends who were always beside me, encouraging me to excel and improve myself.

Abstract

Cybercrime is an evolving and growing threat that heavily bothers Internet users and the relevant authorities. Due to the rapid adoption of technology, Cybercrime Incidents have been increasing at an enormous rate. Cybercrime has several types and various targets, depending on what the offender wants to achieve. It is well-established that the prevention and confrontation of such incidents is a challenging problem since they have a highly complex nature that is constantly developing. This thesis aims to propose a Cybercrime Classification System (CCS) that automatically classifies Cybercrimes into a specific type/class. This process will help to group similar incidents together, propose appropriate counter-measures, design an effective response policy and find recurring patterns between the incidents.

The CCS consists of three Components and each serves a different purpose. In order to classify Cybercrime Incidents, the Cybercrime Classification System uses a feature-based approach, which means that each incident is characterized by some specific, distinctive features. These features will determine the Cybercrime Class that the incident falls into. Machine Learning techniques, such as Data Mining, are used by the CCS for the Classification process.

The results of the presented system are decent and prove that a practical and real-world system which uses the proposed approach could be developed, and it could be proven critical for mitigating Cybercrime. The CCS could be extended in order to provide more functionalities, such as automatically assess threat severity or automatically apply the respective counter-actions.

Key words and phrases: Cybercrime, Data Mining, Classification, Cybercrime Incident Architecture, Cybercrime Classification System

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Topic description

As the Internet and computers have become part of people's daily lives, Cybercriminals are seizing the opportunity to act maliciously. Cybercrime cases have increased dramatically in recent years and criminals are constantly finding new methods to commit Cybercrimes using technology. Cybercrime has many forms: hacking, cyberbullying, sharing illegal material (such as child pornography), credit card fraud etc. It is a fact that the various offenses of Cybercrime constitute a serious threat to the general safety and economy of the global society. It is now more necessary than ever for this threat to be addressed by the authorities in order to protect potential victims, but some obstacles make it difficult to deal with this phenomenon. Cybercrime's complex nature incommodes the authorities since they have many challenges to overcome, but it should be highlighted that important progress has been made towards handling Cybercrimes.

A significant contribution that would assist the mitigation and confrontation of Cybercrime Incidents is *Classification*. Classification (or categorization) refers to the process of classifying a Cybercrime Incident into a specific category, based on its characteristics, such as the offender or the access violation. This process would help expert analysts to search for correlations between Cybercrimes, to generate helpful insight or statistical data and also to automatically propose the appropriate response measures to the relevant offense according to its Cybercrime category.

This thesis presents a novel method to classify such incidents via a system which

consists of three distinct components. Each component is responsible for different processes that need to be executed in order for the Classification to work. With the use of special Data Mining software, this procedure can also be applied to real-world situations and help the authorities to handle Cybercrime more efficiently.

1.2 Aim & objectives

The aim of this thesis is to present a practical approach towards the categorization of Cybercrime and develop a framework that will significantly contribute to solving a severe and growing problem by combining the Cybercrime domain with Data Mining techniques. Specifically, the use of an automated Classification System can help the authorities to propose the appropriate counter-actions in order to handle the Cybercrime Incident (CI), to monitor, analyze and handle similar occurrences, to assess the threat severity of each incident, to identify possible correlations between the features of the CI, to produce data that can be analyzed, for example, in terms of frequency, and therefore prevent such incidents.

This thesis attempts to classify Cybercrime Incidents based on their features by using the proposed Cybercrime Classification System (CCS). CCS uses Machine Learning techniques such as Data Mining and Classification to categorize the offenses. In order to do so, it is required to systematically generate synthetic data that is basically a set of Cybercrime Incidents that will help the CCS to ”learn” correlations between Cybercrime’s features so that it can automatically categorize new, unknown instances to a specific category.

A major purpose of this thesis is to present new methods that could be applied in the real world in order to assist in the mitigation of Cybercrime, and also to be a possible inspiration and foundation for other researchers who are interested in the Cybercrime domain. The general philosophy of the proposed system could possibly be used by other architectures that focus on handling Cybercrime Incidents through a holistic approach.

1.3 Methodology

This thesis has its core foundations based on the paper *A Cybercrime incident architecture with adaptive response policy* (Tsakalidis et al.) [1]. This paper proposes a complete architecture for handling Cybercrimes and introduces the significance of the Classification. It presents the idea of Cybercrime Classification from a theoretical perspective. This thesis expands the functionality of the said architecture by suggesting a practical approach to the Classification problem.

For the completion of the proposed Cybercrime Classification System, the author had to create data in a systematic way since there are no publicly available data regarding Cybercrime Incidents. The author presents a systematic process for generating synthetic data that is required in order to train the proposed system and to automatically predict the category of new incidents.

Special software *(Weka)* is used for the required Data Mining processes and also a programming script is used for generating the said synthetic data. The respective pseudocodes are presented in order to simplify the relevant procedures.

This thesis thoroughly presents the system's functionalities and explains the logic behind each process. Experiments are performed in order to validate our assumptions and, in combination with a case study, the significance of the proposed system is made obvious. The evaluation of the results is made by using specific metrics (e.g. accuracy) and also by manually running algorithms in order to compare the results.

In conclusion, the development of the proposed Cybercrime Classification System had a series of obstacles, such as lack of existing data, creating a proper dataset, and missing information regarding the correlations between Cybercrime's features. However, by combining already existent research methods and by proposing new techniques, this thesis manages to achieve its target which is to provide a practical approach for classifying Cybercrime Incidents.

1.4 Structure

Chapter 1, the current Chapter, describes the main topic, the purposes and the methodology that is used for conducting this thesis research.

Chapter 2 discusses the theoretical background of this thesis and thoroughly presents

the two main topics: Cybercrime and Data Mining. It presents the growing threat of Cybercrime and its characteristics and the most common Data Mining techniques used to gain insight from data.

Chapter 3 reviews and summarizes the paper *A Cybercrime incident architecture with adaptive response policy* (Tsakalidis et al.) [1]. Since this thesis has its foundations on the said paper, it is critical to understand the basic ideas behind the initially proposed architecture and how this thesis expands its functionalities.

Chapter 4 presents and proposes a novel Cybercrime Classification System that utilizes, among others, Data Mining techniques, and aims to classify Cybercrime Incidents into predefined categories. This system heavily contributes to the automation of Cybercrime handling, including proposing the appropriate counter-measures, finding recurring patterns among the incidents etc.

Chapter 5 concludes this thesis by presenting the practicality of the Cybercrime Classification System and its contribution to Cybercrime mitigation. It also mentions the possible future research that could be done to enhance, improve and take advantage of the proposed system.

Chapter 2

Theoretical framework

2.1 Cybercrime

Cybercrime is a relatively new criminal act that is the product of the recent evolution of Information Technology (IT). Today, the internet is a crucial aspect of daily life for most people. As part of their everyday life, most corporations, countries and individuals depend on this new technology to get things done. Computers have changed the way people work, interact, and socialize. Every day, many individuals use the internet to conduct business, conduct research, collect data, shop, entertain, perform banking and financial tasks, transfer files and data to others, and connect with friends all around the globe.

It is undeniable that technology has contributed heavily to social evolution and has made many tasks simpler for people. Science has made a quantum leap during the last 10 years, communication between people is easier than ever, remote work is usual in office jobs via technology mediums (which is something we saw intensely during the COVID-19 crisis).

However, the heavy adoption of technology has created many new opportunities for devious people aiming to benefit themselves by harming others: the so-called **cybercriminals**. Illegal online activity is a daily phenomenon and has been proven a quite serious threat for most computer users, especially the ones who are not very familiar with the technology. Apart from the fact that new crime types have erupted, some traditional crimes have been altered to make use of technology. For example, money laundering is getting done using cryptocurrencies (such as Bitcoin), cyberbul-

lying is getting done via social networks and messaging applications (such as Facebook or Snapchat) and so on.

The convenience of these Cybercrimes has made technology appealing to malicious people; even one unskilled individual can make use of specific software and hack into vulnerable systems or even steal someone's credit card details. Anonymity is also an important factor especially regarding cyberbullying or harming someone's reputation for revenge, or even for vanity reasons.

2.1.1 Definition

According to Casey [2], **Cybercrime is when a crime incident involves computers and networks**. There are however some misconceptions about cyberattacks and Cybercrimes which will be analyzed in the next sections. In general, Cybercrime is any criminal activity that involves digital mediums and technology such as computers, smartphones, networks aiming to harm other individuals or corporations, or even an ICT Infrastructure. It has many types: hacking, cracking, misuse of devices, alteration of data, identity theft and so on. Apart from these, some Cybercrimes involve some forms of violence: harassment, cyberbullying, child pornography.

The European Commission claims [3] that "the term Cybercrime is applied to three categories of criminal activities. The **first** covers traditional forms of crime such as fraud or forgery, though in a Cybercrime context relates specifically to crimes committed over electronic communication networks and information systems (hereafter: electronic networks). The **second** concerns the publication of illegal content over electronic media (i.a. child sexual abuse material or incitement to racial hatred). The **third** includes crimes unique to electronic networks, i.e. attacks against information systems, denial of service and hacking.". We can see that Cybercrime includes a wide range of offenses and it is multidimensional, making it difficult for authorities to prevent and respond to every Cybercrime incident with unqualified success.

The US Department of Justice has distinguished three main Cybercrime types [4]:

1. Crimes in which digital devices, such as computers and networks, are the main target of criminal activity. For instance, hacking, malware or DDoS attacks.
2. Crimes in which digital devices are just the tools for executing the crime. They serve as the medium but are not the target. For example, child pornography,

cyberbullying or identity theft.

3. Crimes in which the use of the computer is ”an incidental aspect of the commission of the crime but may afford evidence of the crime”. For example, emails and documents found in the computer of a murder suspect, or online conversations of a suicide victim. In such cases, digital devices are not really involved in the actual crime but may provide evidence. Digital Forensics plays a major role in these crimes.

This thesis focuses on the first two types since the third one is mainly about Digital Forensics and the examination of various evidence.

Another criterion for categorizing Cybercrimes could be violence. We can divide Cybercrime into two broad, general categories: *Violent* and *Non-violent Cybercrimes*:

1. **Violent Cybercrimes**: Violent Cybercrimes are crimes that involve violence in any form. With the rise of Social Media and with more and more kids and teens using the Internet, it is considered easy to commit a violent crime using digital mediums. The most common forms are *cyberbullying*, by exposing embarrassing conversations or pictures, or by verbally assaulting the victim, *sharing pornographic material without permission* (revenge porn) which is a form of psychological violence and has become very common in the last years. These crimes have the **human** as the target and have a major impact on the sanity of the victim.
2. **Non-violent Cybercrimes**: Non-violent Cybercrimes usually have financial motives. The most common incidences include attacking an ICT Infrastructure of a competitor company for personal benefit, performing Identity Theft of an individual in order to steal money, stealing credit cards, sharing copyrighted material for download without the permission of the owners and so on. However, sometimes hacking, phishing and these type of attacks happen without financial motive but just for entertainment from the so-called ”script kiddies” which make use of existing powerful tools. These crimes, although they do not target humans directly, have an indirect impact on them since they lose money, time and trustworthiness (if the target is a company/organization).

Understanding the difference between Cybercrimes and Cyberattacks — It is common that the terms **Cybercrime** and **Cyberattack** are used interchangeably. There are however some crucial differences that have to be highlighted. The main difference is that Cybercrimes have the **human** as a target, either directly or indirectly, while Cyberattacks are mainly focused on the computer as a target. Cybercrimes use technology as a tool to harm others. The literature does not have efficient information on this important detail. Roderick Graham attempts to distinguish cybersecurity (which is tightly associated with cyberattacks) and Cybercrime [5]:

Differences between the Concepts of Cybersecurity and Cybercrime

Cybersecurity	Cybercrime
• Applied science oriented – coding, networking, and engineering strategies for making networks more secure	• Basic science oriented – theoretical understandings of how and why crime is committed
• STEM disciplines – computer science, computer engineering, information technology	• Social science disciplines – criminology, psychology, sociology
• The primary law enforcement bodies are federal	• The primary law enforcement bodies are state and local
• The victims of interest are government and corporate networks	• The victims of interest are individuals
• The crimes are more focused on computer as target (malware, SQL injection, Direct denial of service attacks)	• The crimes are more focused on computer as tool (identity theft, romance scams, cyberbullying, fraud, hate speech)

Figure 2.1: Differences between Cybercrimes and Cyberattacks [5].

Defining Cybercrime is challenging since it is still developing and the world is still learning about it. Although the term Cyberattack is strictly tied with Cybercrime, we must understand that they do not represent the same thing and the differences, as presented above, must be taken into consideration.

2.1.2 Historical & statistical data

Historical data — Regarding Cybercrime, some events have been crucial in this field. Cybercrime's history is presented in this section to illustrate how Cybercrime has developed over the past few years.

In the 1970s, technology began to grow at a satisfactory rate. Although until then the use of computers was limited in big companies and the US military and American universities using ARPAnet, cybercriminals had begun to become familiar with the vulnerabilities of computers. One of the first viruses, "Creeper", was transmitted over ARPAnet and notified users that their systems have been infected[6]. In general, at that time computers were not used widely so usual people did not fully understand the dangers of Cybercrimes and cyberattacks.

By the 1980s, computers had become quite mainstream. People were using the Internet more often and businesses were taking advantage of its handiness. Cybercriminals were a step ahead of the public and saw the possible benefits of exploiting the system's vulnerabilities for their own advantage. "Hacking" became a popular word and new viruses were showing up, cybercriminal groups were established and people realized that a new threat was evolving. In 1981, the extensively known viruses "Apple 1, 2 & 3" began to spread among systems which used Apple II operating system. Texas University was the organization in which the virus was first detected [6]. Most users who had downloaded pirated games or software were infected by the Apple virus.

The 1990s was a very crucial period for Cybercrime and computers, in general. During this period, Personal Computers (PC) were becoming common. Almost all countries and big corporations had had computers used regularly for their tasks. It is obvious that the Internet was used by more and more people in their everyday life, which is why cybercriminals were also starting to committing more Cybercrimes. It was easy, for example, to send a malicious file using the AOL messaging software which was widely used for everyday communication back then, and steal their possibly confidential files.

In the 2000s, PCs and the internet were almost in every home. People were using them for work, for communication, for entertainment and for studying. Cybercrime was also at a very high growth rate. Countries, companies, the public were all targeted for various reasons: social harm, economic harm, infrastructure harm etc. At the time,

the need for antivirus software was intense. Cybercriminals started to face more difficulties because of the antiviruses, but this did not stop them from harming systems or other people.

In the 2010s, the cyber world saw quite a few severe breaches and cyberattacks beginning to threaten national security and business. The increasing adoption of smartphones and digital devices made cybercriminals more eager to attack. One of the most important attacks of the said decade was the WannaCry ransomware which infected over 300.000 computers and harmed over 200.000 victims [7]. This virus was encrypting all the system's files and was asking for payment in BitCoin cryptocurrency in order to decrypt the files and not delete them permanently. On the other side, cybersecurity was also developed in order to tackle the various attacks and played a major role in protecting countries, businesses and common people.

As we experience the 2020s, technology has taken over our lives. It is a basic tool in almost every aspect of our everyday life and this became obvious when the whole world went through unexpected situations due to the COVID-19 pandemic which forced people to work remotely using technology mediums. Recent researches have shown that the COVID-19 pandemic increased cyberattacks and Cybercrimes incidents [8]. It is worth noting that emerging technologies such as 5G networks, Internet of Things (IoT), Smart Cities and others, are also going to dramatically change our lives in the next years. Additionally, the rise of FinTech (Financial Technology) sector has pushed people to increase their financial transactions via the internet and therefore expose them to a variety of dangers. As a result of these developments, Cybercrimes and internet attacks are expected to grow exponentially in the next few years.

Statistical Data — As Cybercrime is evolving and expanding, it is important to study the relevant statistics. Cybercrime has become highly complex and sophisticated in the last few years and the fact that people and companies use the internet more and more, the risks have never been more serious.

According to Statista [9], the average cost of a data breach worldwide is 3.86 million US dollars, whereas about 51% of the organizations who are victims of a cyberattack choose to pay the ransom after a ransomware attack. This is why companies and organizations are constantly trying to improve their cybersecurity and invest in protecting their ICT Infrastructure.

The exponential growth of Cybercrime incidents is obvious in the below chart which shows the registered cases of Cybercrime in Belgium from 2008 to 2019:

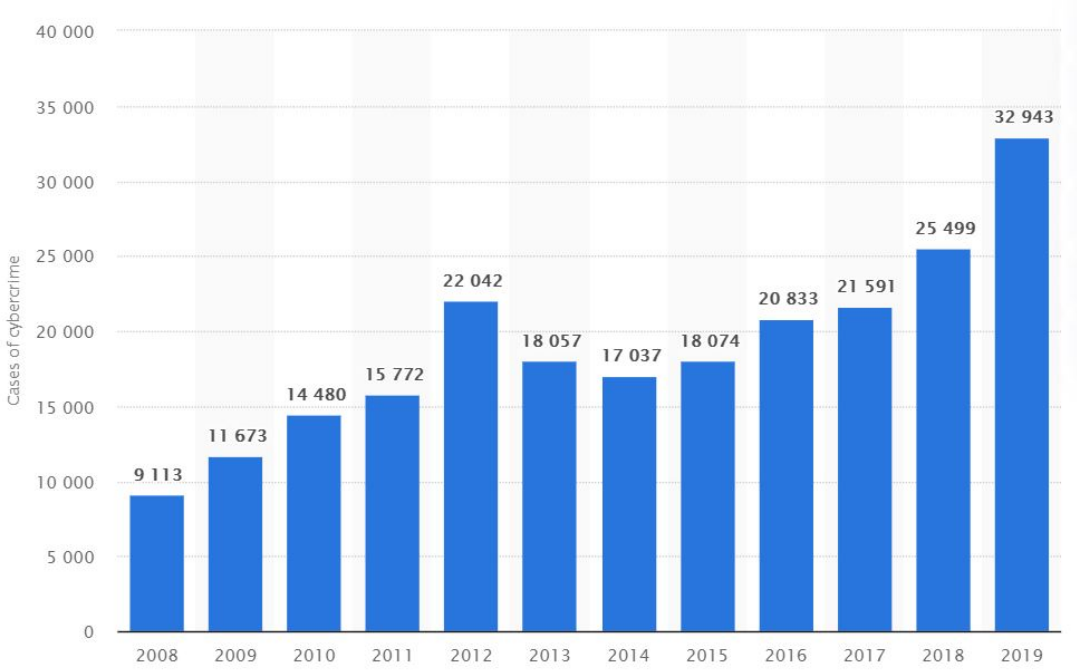


Figure 2.2: Registered cases of Cybercrime in Belgium from 2008 to 2019. [9]

Equivalent increasing rates apply to almost every country in the world, and some countries such as Russia, China and India face even more serious problems. Note that these numbers only represent a small portion of the total Cybercrime Incidents that really occurred, since a significant percentage of individuals choose not to report Cybercrimes to the authorities [10] because they lack knowledge, they believe that it is not that important or that it will not help them recover their funds, their integrity or resolve the problems that it caused.

2.1.3 The challenges of Cybercrime

In one of the most influential papers for analyzing crime rate trends and cycles [11], it is stated that the commission of a (traditional) crime requires three basic factors: 1) supply of motivated offends, 2) availability of suitable opportunities, 3) absence of capable guardians. The same applies to Cybercrime; the digital world we live in fulfills all these three requirements.

Due to its complex and sophisticated nature, Cybercrime has many challenges which authorities and the world in general, need to overcome. Although the research

community and social or computer scientists have made significant progress, the below characteristics [12] render Cybercrime difficult to mitigate and control.

Scale — Unlike more conventional forms of communication, the Internet enables users to communicate with a large number of individuals cheaply, effortlessly and quickly, let alone the productivity tasks that an individual can perform with the use of computers. It is estimated that 4.66 billion people use the Internet [13], as of January 2021. This translates to about 60% of the world population. This tremendous number of people means that there endless possible victims, but also endless possible offenders and cybercriminals. This phenomenon is very convenient for malicious people; it allows them to commit crimes on a scale that is not possible in the real world. The ability to automate cyberattacks and Cybercrimes using special tools magnifies this effect.

Accessibility — A few years ago, computers were too expensive and were used only by big organizations and governments. During these times, the chance to commit a Cybercrime was limited since only a few people had access to computers and they had to have very good technical knowledge in order to do so. Nowadays, almost every household has a computer connected to the internet; this means more possible victims and cybercriminals.

Anonymity — The anonymity that the internet provides makes Cyberspace a very attractive place for criminals to commit crimes. The ability to trace back attacks is becoming more and more difficult due to many technical ways that exist in order to hide the identity of a user: proxy, VPN, encryption techniques and others. Anonymity is the first priority of an offender and he always makes sure that evidence does not point back to him. The complexity of networks and computer systems makes the work of the authorities difficult and time-consuming. Anonymity is the biggest concern regarding Cybercrime since it contradicts with user's privacy and therefore it is challenging to find a balance between them [14].

Absence of guardians — In the "offline world" when a criminal plans to commit a crime, he always takes into consideration the perceived risk of getting caught and prosecuted. The same applies to Cyberspace but with some important differences. Firstly,

digital evidence, not only is sophisticated to collect with Digital Forensics techniques, but also it is challenging to present them as evidence for use in a criminal trial [12] since digital evidence can always be altered and therefore question its integrity and validity. Secondly, cybercriminals know that it is difficult to constantly monitor cyberspace (just like patrols are monitoring in the real world) due to its enormous size. It is obvious that, since cyberspace cannot be monitored and controlled, the users must take their actions and defend themselves with all the available ways. Whether this means that parents teach their kids to never share private information online, or that every user uses an antivirus software, it is mandatory to always be watchful for numerous threats while using the internet.

2.1.4 Cybercrime features

Cybercrime is usually characterized by great complexity in terms of its particular characteristics. It is therefore a logical outcome that the terms which define Cybercrime are difficult to understand and often overlap. For this reason, it is necessary to clarify, according to the literature, the features which can define a Cybercrime Incident. A thorough explanation of these features is given in the paper *Tsakalidis, G. (2016). Camco a framework for classification, analysis & monitoring Cybercrime-related offenses.* [15]. We will try to give a more brief explanation based on the forementioned paper and we are going to exclude some of them in order to adjust them to our topic. These features, as presented below, are going to be utilized by our *Cybercrime Classification System* which is the main focus of this thesis.

Offender — Offender refers to the individual who is responsible for committing or participating in a criminal offense (Cybercrime, in our case). Offenders can be divided into many types, such as cybercriminals, script kiddies, abusive users and others.

Access violation — Access Violation refers to the way which was used to commit the Cybercrime. It answers the question of **how** the incident took place; did the criminal had physical access to the computer? Did the criminal gain access to the victim's computer through a virus? Did the criminal caused physical harm to a network?

Target — Target is about the main purpose and the motive behind the Cybercrime: why did the subject commit the crime? Did he want to cause financial, emotional or social harm? The answer to this question is critical because it can show where Incident Response should focus the most.

Victim — Victim is about the Individual, the Company/Organization or even the Country/State which has been damaged or suffered as a consequence of the Cybercrime Incident.

Harm — Harm is about the results of the CI. What harm did the offense cause? Infrastructure, Social, Individual or Inchoate damage?

Attribute values

Each of the abovementioned features has a specific set of attribute values. These values are presented and explained below in order to understand the underlying meaning of each value [15].

Offender

- *Abusive user*: Offender who propagate hate speech, glorification of violence, verbal assaults
- *Cyber-bully*: Offender who intend to insult, hurt or embarrass other individuals
- *Cyber-criminal*: Offender who aim to obtain intelligence and profit from illegal internet activity
- *Cyber-fighter*: Offender who is politically motivated and is contributing to cyber-activity of their country
- *Cyber-terrorist*: Offender who has terrorism purposes and uses cyberspace to recruit personnel, communicate with co-members throughout the world and perform cyberattacks against critical infrastructures
- *Hacktivist*: Usually a group of hackers that has media exposure due to their political motivation. Their main purpose is to reveal critical information from or-

ganizations, companies, authorities or politicians in order to expose them. They consider themselves activists who raise public concern.

- *Insider (employee)*: Insiders usually are motivated by revenge, profit, extortion or sabotage [15]. The most usual form of Cybercrimes involving insiders is a critical data breach.

- *Online social hacker*: Online social hackers use a relatively new form of Cybercrime: social engineering. Social engineering exploits certain characteristics of human behavior and uses emotions in order to create a seemingly trust relationship with its victims. The rise of social media has assisted Online social hackers since users choose to expose their personal lives on the internet.

- *Script kiddie*: The term *Script kiddies* refers to teenagers who perform illegal activity using tools widely available in cyberspace. Usually, these Cybercrimes only happen for entertainment purposes and to the naive behavior of young persons.

- *Sexually deviant user*: These offenders use the internet in order to download, distribute illegal pornographic material, including child pornography. It is one of the most commons Cybercrimes. They usually approach kids via chatting applications and they impersonate as a peer.

- *Company/Organization*: This type of offender is about entities that consist of individuals with the same motivations and ideas. They have a common target and they work together in order to reach it.

Access violation

- *Physical tampering*: Physical tampering refers to physically damaging or corrupting hardware and ICT infrastructures in order to destroy or make them malfunction.

- *Local access*: With local access to a computer, an offender can install malware, leak information, perform data breach etc.

- *Remote access*: Remote access is the most common way of access violation since it allows anonymity, speed and access to the whole world without physical restrictions.

Target

- *ICT (Confidentiality, Integrity, Availability*: Offenders may target the *CIA Triad* of computer security, namely Confidentiality, Integrity, Availability.
- *ICT Infrastructure*: Attacks to an ICT Infrastructure aim to interrupt the functionality of computers and networks by abusing them with malicious methods.
- *Social*: Offences that target social principles in order to limit access to social services usually provided by big organizations or countries.
- *Financial*: When offenders aim to financially benefit themselves by scamming or stealing victims (individuals or companies).
- *Emotional*: Cybercrimes that aim to emotionally harm others, usually for revenge purposes.

Victim

- *Individual*
- *Company/Organization*
- *Country/State*

Harm

- *Infrastructure*
- *Individual*
- *Social*
- *Inchoate*

These features are thoroughly defining each Cybercrime occurrence and this process will assist the seamless execution of the proposed Cybercrime Classification System, which will be presented in the next Chapters.

2.1.5 Cybercrime prevention

As technology is constantly evolving to fight the growing Cybercrime threats, human behavior comes to complicate things. As it has already been stated, Cybercrime is a complex domain and therefore it is also complex to develop a very effective and reliable strategy to prevent it. Many factors allow Cybercrime to grow: human factors, technical factors, lack of knowledge, naivety etc. In this section, the main factors will be presented in order to provide the main ideas which must be implemented for mitigating and preventing Cybercrime incidents.

End-user protection — Each computer is vulnerable to a wide variety of threats: viruses, ransomware, trojans, phishing, keyloggers and others. The constant exposure of the computer to the internet makes Antiviruses necessary. Antiviruses have improved over the last years and have managed to protect users from an enormous number of threats. Users must always be aware of the source when downloading a file (torrents are often malicious), they must always pay attention to the site's URL in order to verify the authenticity of the website so they can avoid phishing attacks, for example when visiting a bank's website.

In conclusion, users should always be aware of the available threats and they should carefully think when they are acting, such as downloading a file, executing an unknown program etc. By using a reliable Antivirus software, such threats can be easily prevented, automatically.

Education of children/teenagers and elderly people — As more and more younger people use digital devices and the Internet, it is necessary more than ever to educate children and teenagers regarding cyberspace's threats. However, from the other side, elderly people who are in the first steps of using the Internet are also quite vulnerable to its threats. Young people are characterized by naivety and elderly people are characterized by lack of knowledge.

Since young people are more naive than mature users, parents should educate themselves and apply the appropriate techniques to protect their children. Parents should raise awareness for the danger of anonymity, since pedophiles and sexually deviant users target children via Social Networks and they act to be a peer, with the ultimate goal being to meet them person-to-person in order to harm them or molest them.

On the other side, the digital transformation of Countries has made necessary the use of the Internet. Therefore, although they do not desire to, elderly people have to use computers. Since they are inexperienced, they are very vulnerable to expose their personal information to the wrong place, or they can easily infect the computer. Educating elderly people regarding technology is a crucial factor for preventing Cybercrimes from happening.

Human factors — As Back and LaPrade state in their paper [16], human behavior is a factor that cannot be predicted since people can always be vulnerable while they use technology mediums such as the Internet and they can always take unpredictable actions.

For the above reason, it is reasonable to take a more holistic approach and carefully study the human factors which affect Cybercrime so that scientists can develop prevention policies more effectively.

Researchers have concluded that demographic factors, social context and victimization experiences are closely related and affect the public's fear of Cybercrime on Social Networks (e.g. Facebook, Twitter) [17]. The paper highlights the fact that human interaction and behavior in Social Networks impact the topic of Cybercrime victimization. It is, therefore, useful to design prevention policies based on human factors.

Technical factors — Of course, individuals but also companies and countries should take preventive technical measures. This includes, but not limited to, the below actions:

- Keeping software updated
- Using an Antivirus software
- Using Firewall
- Regularly scanning for infections
- Design of security policies for the proper use of specific programs
- Using encryption for critical documents
- Using two-factor authentication for logging in to websites
- Keeping the Operating System updated

- Close unused ports
- Not keeping sensitive information, such as PIN codes, bank accounts etc. on the computer

Information Technology scientists should always devote time in order to secure the company's systems and prevent attacks from happening than trying to confront them later on. Common users should also take their basic measures for protecting their PC as highlighted above.

2.1.6 Cybercrime management strategy

In order to confront Cybercrime, we must develop a Cybercrime management strategy focusing on many aspects. A legislative framework is needed more than ever, and along with technical strategies, law enforcement agencies should be able to mitigate Cybercrime at a significant level.

Focus will be given to five methods that will form an effective Cybercrime management strategy as shown in the below figure:

Figure 2.3: Cybercrime management strategy.

Legislative and institutional framework — As Cybercrime is evolving, it is needed to strengthen laws for controlling the Cybercrime rates on the Internet. A strict legislative framework should be introduced instead of general guidelines, in order to prevent and demotivate cybercriminals from committing the crime.

It is a fact that Cybersecurity is critical for national security since many Cybercrime occurrences target critical infrastructures of countries which may lead to data leaks or severe damage. For this purpose, public and private sector cooperation is necessary, as they should share cybersecurity information which will form the relative legislative framework.

Awareness — Spreading awareness is a simple method for preventing Cybercrime as a significant percentage of Cybercrimes happen due to the lack of awareness [18]. Basic knowledge can reduce digital crime such as Social Engineering, phishing etc.

It is important to organize public awareness campaigns in order to inform people about the possible threats and also educate them on how to tackle them. It should be highlighted that the reporting of such incidents is very crucial for the authorities so that they can design a more effective strategy, adjusted to the real-world data.

Big data techniques — Big data is a trending topic in today's digital world. Nowadays, data can be found everywhere: cameras, sensors, the World Wide Web and so on. Therefore, analysts could use this data in order to apply big data & data mining techniques.

Collecting and analyzing Cybercrime data could be applied so that authorities gain intelligence that could help mitigating Cybercrime incidents. Finding correlations and patterns between cyberoffenders and between incidents could also assist the prediction of similar occurrences.

Researchers have recorded four basic big data modeling techniques aiming to tackle Cybercrime [19]:

1. *Predictive*: Analyze current and historical facts in order to predict future Cybercrime incidents.

2. *Descriptive*: Analyze Cybercrimes and identify the relationships between factors that are responsible for them.

3. *Diagnostic*: Identify why a Cybercrime incident occurred by analyzing historical data and examine evidence to identify the probable causes.

4. *Prescriptive*: Use data in order to develop and improve policing and monitoring strategies that will lead to the prevention of Cybercrime occurrences.

International collaboration — Countries, organizations and companies should work together in order to effectively tackle Cybercrime. FBI, Europol, Interpol and Intelligence agencies around the world should share information so they can combine intelligence and form a Cybercrime management strategy that will diminish digital crime.

Technical enhancements — As cybercriminals develop new ways to commit crimes, law enforcement agencies need to enhance their technical capabilities and catch up with them. New threats are constantly showing in cyberspace and therefore new confronting methods must be developed.

Researchers point that ”it is important to make a transition from working in isolation to a collaborative approach and increase their capabilities through technical empowerment of their cadre” [19].

2.2 Data mining

2.2.1 Definition

Data Mining is the process of collecting, processing, analyzing and gaining useful insight from data [20].

Nowadays, data are found everywhere and are about anything: internet users, financial interactions, sensors and IoT, social media and so on. Almost every aspect of a person’s life is on the internet. The amount of data available online is enormous and this is why Data Mining could be proved very useful for gaining information regarding a wide variety of things.

However, in most cases, the raw data are unstructured and cannot be processed directly. In other words, data must be processed in order to be useful and understandable for humans and machines. Preprocessing is a critical task of the Data Mining workflow and is about collecting, cleaning and transforming the data into a structured format.

Data, in its primary form, have different formats: it could be quantitative (e.g. weight, money, age), categorical (e.g. countries, animals), text (e.g. words of a book) and others. Each Data Mining problem has its own challenges to overcome in terms of preprocessing and analyzing the data.

In general, Data Mining process involves 3 distinct phases:

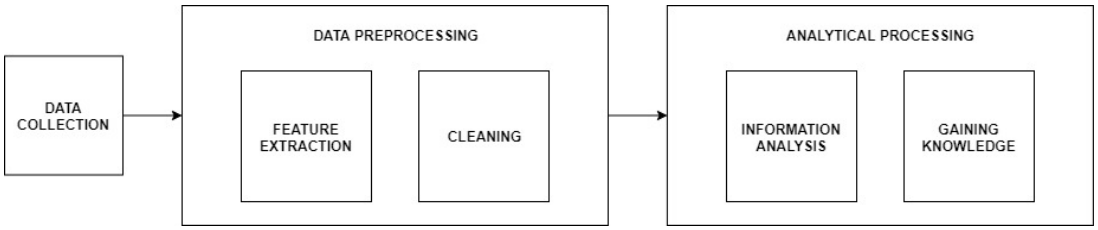


Figure 2.4: Data Mining phases.

1. *Data collection*: Data collection is about collecting and extracting the available data from the corresponding source. The source could be a set of sensors, a database, the World Wide Web or the search engines along with a web scraping tool. This phase is very critical for the execution of a decent Data Mining process since the selection of proper data has a significant impact on the generated insight.

2. *Data Preprocessing*: Data Preprocessing involves Feature Extraction and Data Cleaning. As we stated before, when the data are first collected, they are often in a form that is not appropriate for processing. Therefore, it is required to extract the features from the data which will be useful for our Data Mining process. Features are some properties that can be extracted from the data and provide us some useful information. After extracting the features, usually, we have to preprocess the data in order to give them a more structured form, or we may want to remove duplicates or missing values. After the execution of these processes, the data has now a structured and consistent form, ready to be given to the Data Mining technique.

3. *Analytical Processing*: The final step of the Data Mining process is to apply analytical methods in order to gain insight. Will we use Classification or Clustering? How will we interpret the Data Mining results? Is it meaningful to apply Classification and then Clustering? Can these data be joined with other data in

order to produce more insight? These are some questions that must be answered in order to complete the Analytical Processing phase successfully.

2.2.2 Data Mining techniques

Data Mining has several techniques with the most important being: *1) Association Rules, 2) Clustering & 3) Classification*. An overview of each technique will be presented in order to understand the respective use cases and characteristics.

Association Rules

Association Rules is a Data Mining technique that produces rules in the form of *If/Then* statements which help discover relationships and associations between data [20]. It aims to create rules that detect the frequent patterns or correlations from various kinds of data. This technique is only suitable for categorical data and not numerical.

An association rule has 2 parts:

- a*antecedent (if)*
- *consequent (then)*

The antecedent is an item that is found in the data while the consequent is an item that is found combined with the antecedent.

One of the most popular areas where Association Rules are applied is in Super Markets; the so-called Market Basket Analysis. For instance, let's assume that the following rule is found [21]: *"If a customer buys bread, he is 70% likely of buying milk"*. In this example, *bread* is the antecedent and *milk* is the consequent. Stores can use this Association Rule to target the consumers more effectively and of course to increase their revenue.

AR are calculated from the so-called *itemsets* which are made up of two or more items. The rules are produced by analyzing data with software, and specifically by looking for frequent *If/Then* patterns among itemsets. In order to define which patterns are frequent, AR uses two important metrics to calculate the strength of a given association rule:

1. *Support*: Indicates how frequently the *If/Then* relationship occurs.

2. *Confidence*: Indicates the number of times the *If/Then* statements are found true.

For example, given the rule $X \rightarrow Y$ we can calculate Support and Confidence as below (*frq=frequency*):

$$\text{Support} = \frac{frq(X,Y)}{N}, \text{ Confidence} = \frac{frq(X,Y)}{frq(X)}$$

Often a third metric is used, known as *lift*, which is the ratio of confidence to support. If the lift value is negative, then there is a negative association between the respective data whereas if the value is positive there is a positive correlation. If $lift = 1$, there is no correlation between the data. Its formula is the following:

$$\text{Lift} = \frac{\text{Support}}{\text{Support}(X) \times \text{Support}(Y)}$$

In terms of algorithms, AR uses *AIS, SETM, Apriori* and others. *Apriori* is probably the simplest one, and according to Lutkevich, [22] it works like this: the Apriori algorithm generates candidate itemsets using only the large itemsets of the previous pass. The itemset of the previous pass is joined with itself to produce all the itemsets which have a size larger by one. Now, each generated itemset that has a subset that is not large, is deleted. The remaining itemsets are now the candidates and the algorithm repeats the same process. Apriori assumes that each subset of a frequent itemset is also a frequent itemset and this helps to reduce the number of candidates.

Clustering

Clustering refers to a data mining technique that identifies similar groups of data in a dataset. It divides the data points into a number of groups such that the data points belonging to a cluster have similar characteristics [23].

Clustering has several real-world use cases and researchers have attempted to present its usefulness in certain domains:

- *Time Aware Web Users Clustering [24]*: Group similar web users together based on usage patterns derived from users' preferences. This process produces insight regarding users needs and preferences that could be used for marketing purposes or for bulding user profiles.

- *Using Clustering for Message Scheduling in Networks [25]*: Clustering could help the message scheduling for WDM star networks by creating groups of nodes whose messages have the same destination.
- *Clustering-driven Wireless Data Broadcasting [26]*: Using a clustering algorithm for improving the performance of Wireless data broadcasting on a push-based system.
- *Correlating Time-Related Data Sources with Co-clustering [27]*: Detect dependencies and correlations between time-related data elements using clustering techniques.

Clustering algorithms include Density-based Clustering, Hierarchical Clustering, k-Means Clustering and others. K-means is considered to be the most used Clustering algorithm and is going to be explained briefly.

In general, the k-Means algorithm is considered to be easy to implement and computationally efficient [28]. This algorithm aims to detect and group data points that have high similarity between them into clusters. Each cluster has a center (centroid value). K-means uses a distance metric in order to calculate the similarity: the closer (=smaller distance) the data points are, the more similar they are.

Euclidean distance is commonly used as a distance metric for k-Means clustering. The distance of *A and B* is calculated with the below formula:

$$d(A,B) = \sqrt{(x_1 - x_2)^2 + (y_2 - y_1)^2} \tag{2.1}$$

K-means requires two main parameters to be set:

1. Number of clusters
2. Maximum iterations of the k-Means for a single run

The algorithm's steps are the following [29]:

1. Choose k, the number of clusters we desire
2. Calculate the Euclidean distance between each data point and cluster centers
3. Assign the data point to the cluster center whose distance from the cluster center is the minimum of all the cluster centers

4. Recalculate the new cluster center
5. Recalculate the distances between each data point and new cluster centers
6. If no data point is reassigned, stop. Otherwise, repeat the algorithm from step 3.

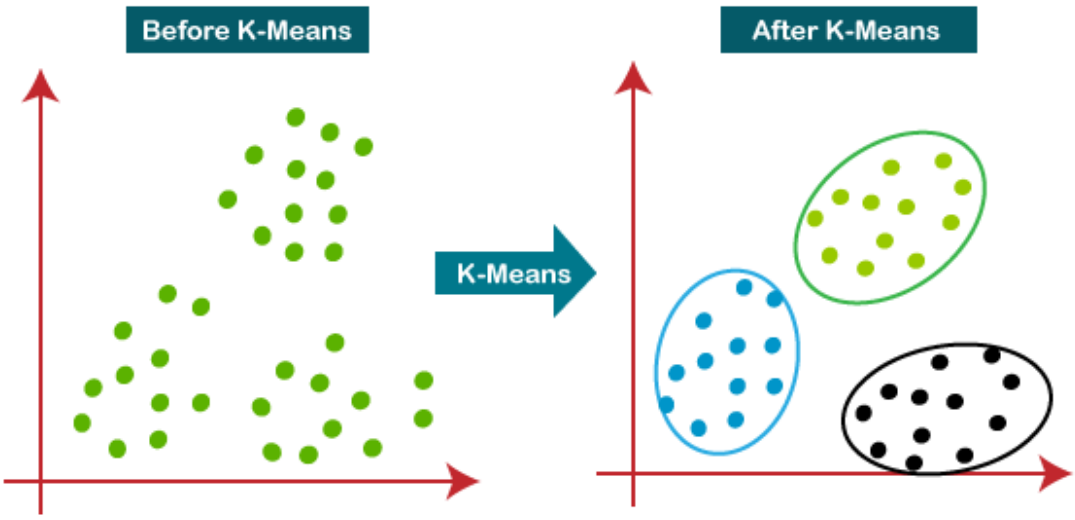


Figure 2.5: k-Means Clustering visualization [30].

Classification

Classification is one of the most popular data mining techniques. The purpose of classification is to predict the class (or class label) for a given (uncategorized) item. A classifier is a model (or function M) which predicts the class for a given input point x, namely: $\widehat{y} = M(x)$, where: $\widehat{y} \in c_1, c_2, ..., c_k$ and each c_i is a distinguishable class label [31].

This technique belongs to the Supervised Learning approach since the model requires in advance a dataset with the correct class label which is called the **Training Set**. The M model learns from the training set and thus can automatically predict the class tag for each given data.

In classification, the rational choice of the data of the learning set is of particular importance, as this will significantly affect the percentage of successful predictions of the classifier. The selection of this data must be done in a thorough way to meet the needs of each problem, otherwise the model risks having a high percentage of incorrect classifications. It is worth noting that although the data size of the learning set must meet a certain limit (quantitative criterion), the **quality** of the data (quality criterion)

has more value for the effectiveness of the model. The quality criteria are different for each classification problem and fall within the discretion of the analyst, but in general, some rules must be followed that are outside the scope of this work.

Classification is the data mining technique that we will be using in the proposed Cybercrime Classification System.

2.2.3 Classification algorithms

Since Classification is the focus of this thesis, we will attempt to present the most important Classification algorithms in order to gain a good understanding of them and, finally, to choose the most appropriate for our Classification problem.

Specifically, the following Classification algorithms will be analyzed:

1. kNN (k-Nearest-Neighbors)
2. Decision Trees
3. Support-vector machine
4. Naive Bayes

kNN (k-Nearest-Neighbors)

The k-Nearest-Neighbors algorithm is a simple supervised machine learning algorithm that assumes that similar data points are close to each other. It is considered a *lazy* method [32] since it does not learn from the training set, but instead it performs the necessary actions on the dataset at the time of Classification.

In order to classify a new data point, kNN calculates the Euclidean distances (see 2.1) of the new data point and the k number of neighbors, it counts the number of data points in each class and it assigns the new data point to the class which the number of neighbors is the maximum. In other words, a new data point is classified based on a majority vote of its neighbors [33].

The kNN classifier follows the below algorithm:

1. Select k value
2. Calculate Euclidean distances of k neighbors

3. Select the k nearest neighbors based on the Euclidean distances
4. Among these k data points, count the number of data points in each class
5. Assign the new data point to the class for which the number of neighbors is the maximum

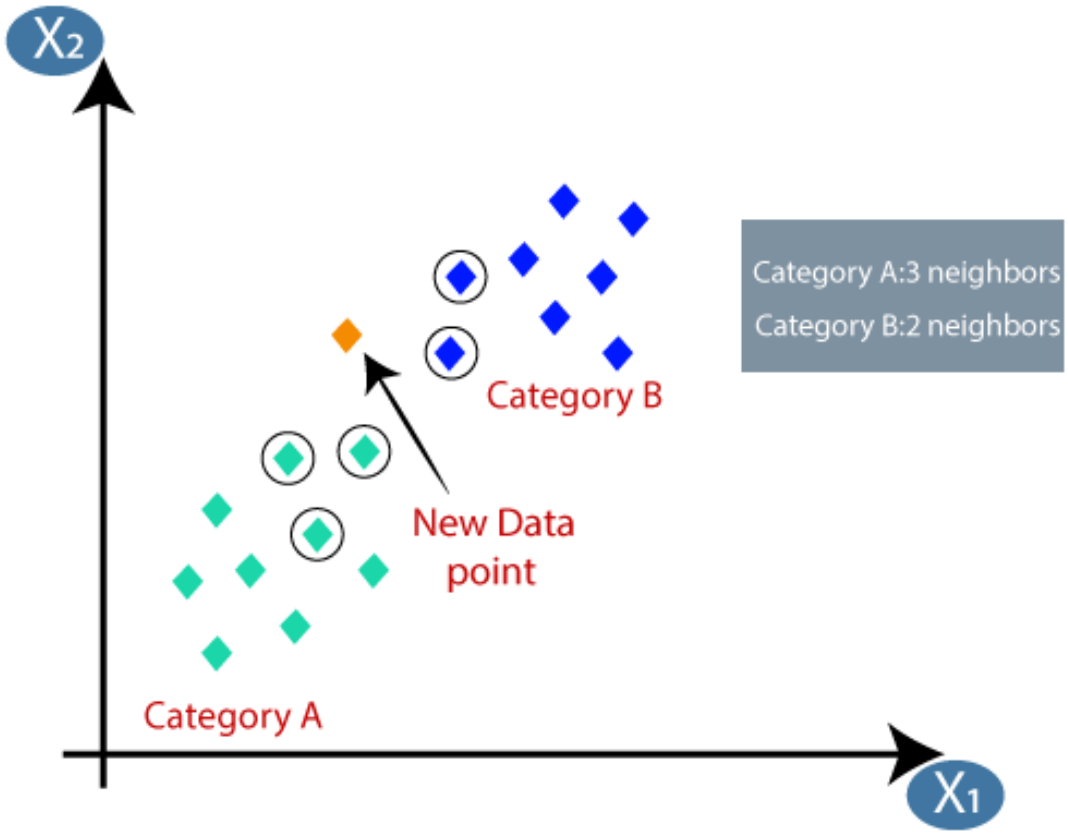


Figure 2.6: kNN Classification example ($k = 5$). The new data point will get classified to Category A since, among the 5 neighbors, Category A has 3 data points instead of Category B which has 2 data points in relation to the new data point [34].

Decision Trees

Decision Trees is a greedy Classification algorithm which uses a top-down approach [35]. It has a tree structure, as the name implies, where internal nodes represent the features of a dataset, branches represent the decision rules and leaves represent the final outcome.

The basic Decision Trees algorithm is ID3 [36] and was originally developed by Ross Quinlan. ID3 algorithm builds the decision tree based on the information ob-

tained from the training dataset and uses the same information in order to classify new instances.

Because every Classification problem differs, Decision Trees allows a set of parameters/criteria to be set in order to optimize/adjust the algorithm to the problem. These parameters are going to affect the decision rules during the execution of the algorithm. The most important criteria are *Entropy, Information gain, GINI index*.

Below is the pseudocode of ID3 algorithm [36]:

Inputs: R: a set of non- target attributes, C: the target **attribute**, S: training data.
Output: returns a decision tree
Start
Initialize to empty tree;
If S is empty **then**
Return a single node failure value
End If
If S is made only for the values of the same target **then**
Return a single node of this value
End if
If R is empty **then**
Return a single node with value as the most common value of the target attribute values found in S
End if
D ← the attribute that has the largest Gain (D, S) among all the attributes of R
{d_j *j = 1, 2, ..., m*} ← *Attribute values* of D
{S_j *with j = 1, 2, ..., m*} ←*The subsets of S respectively* constituted of d_j records attribute value D
Return a tree whose root is D and the arcs are labeled by d1, d2, ..., d_m and going to sub-trees ID3 (R-{D}, C, S1), ID3 (R-{D} C, S2), .., ID3 (R-{D}, C, Sm)
End

Figure 2.7: ID3 algorithm pseudocode [36].

Support-vector machine

Support-vector machine (SVM) is a supervised Classification algorithm which works by trying to find the best decision boundary that can separate data points into classes in order to classify new data points in the correct class. The best decision boundary is called a *hyperplane*. To find the *hyperplane*, SVM chooses the extreme

points/vectors which are called support vectors.

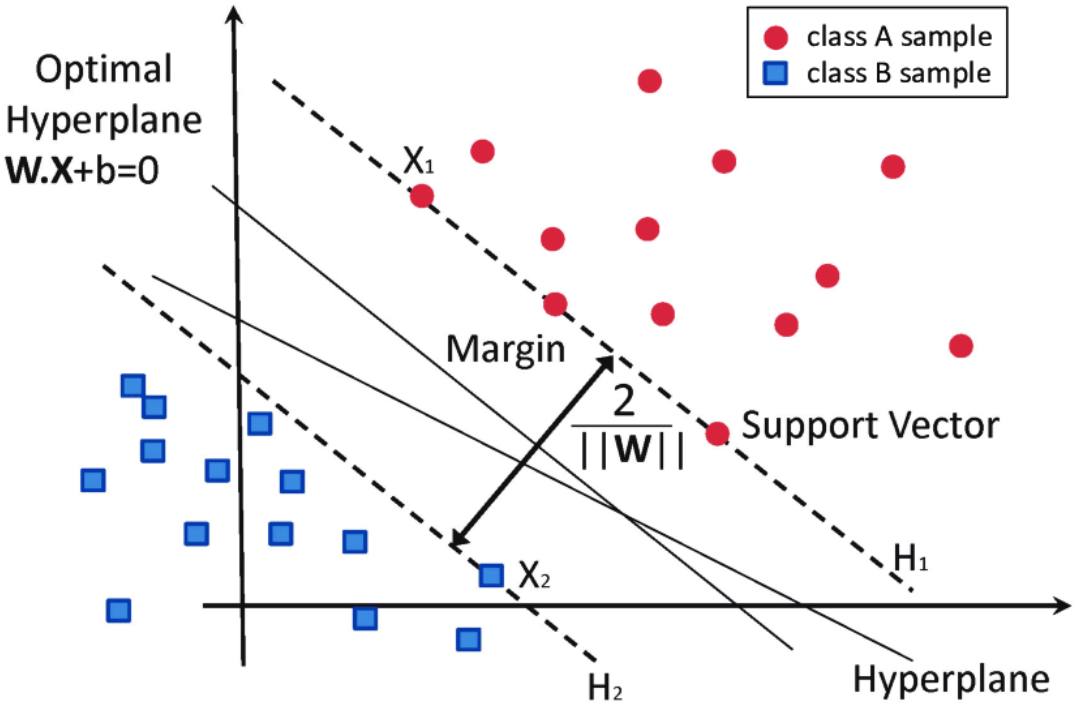


Figure 2.8: Classification of data using Support-vector machine [37].

Hyperplanes are essentially lines which help divide the n-dimensional space into clusters; data points falling on either side of the *hyperplane* belong to different classes.

Although SVM is considered to have high accuracy and good efficiency, researchers point out [38] that tuning SVM is a time-consuming process, as it requires a large set of parameters to be set: kernel functions, standard deviation of the Gaussian kernel, relative weights associated with slack variable and the number of training examples.

Naive Bayes

Naive Bayes is a well-known machine learning classifier based on Bayes' theorem. It is a probabilistic classifier that assumes that each feature contributes to the target class independently and equally. Although NB is a simple classifier, it performs very well on big datasets and has good computational performance. NB classifier calculates the class probabilities and conditional probabilities by processing the training dataset, which are used afterward to define the frequency of each feature value for a given class value divided by the number of instances of that class value [39].

The model — The objective of NB classifier is to determine the probability of the features occurring in each class. Then, the classifier selects the most probable class. For this process, it is needed to calculate $P\left(c_i \mid x_0, ..., x_n\right)$. Bayes' rule is used for the calculations:

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)} \tag{2.2}$$

In classification problems, we can replace A with a target class (c_x) and B with the feature set of our data ($x_0, ..., x_n$). Since for a specific dataset the denominator doesn't change[39], we can ignore it and therefore the class probability is expressed as:

$$P\left(c_i \mid x_0, \ldots, x_n\right) \propto P\left(x_0, \ldots, x_n \mid c_i\right) P\left(c_i\right) \propto P\left(c_i\right) \prod_{j=1}^{n} P\left(x_j \mid c_i\right)$$

Classification — The estimation of the probability of a given data sample belonging to an explicit class can now be done by using the above mathematical approach. Now, the NB classifier needs to handle these probabilities and simply select the c_i which has the highest probability given the sample's features. This is called the **Maximum A Posteriori** decision rule and is express as below:

$$y = \operatorname{argmax} P\left(c_i\right) \prod_{j=1}^{n} P\left(x_j \mid c_i\right) \tag{2.3}$$

Example — Let's assume a dataset with samples such that each sample has three features x_1, x_2, x_3 and one class label c_i, where $i = 1$ or 2. NB classifier has to assign a class label to each sample depending on its features. At first, the algorithm calculates the probability of each class c_i for a specific feature vector (x_1, x_2, x_3):

$$P\left(c_i \mid X_1, X_2, X_3\right) = \frac{P\left(X_1 \mid c_i\right) P\left(X_2 \mid c_i\right) P\left(X_3 \mid c_i\right) P\left(c_i\right)}{P\left(X_1\right) P\left(X_2\right) P\left(X_3\right)} \tag{2.4}$$

By using the proportional rule, as described in *The Model* paragraph, equation 2.4 can be simplified as:

$$P\left(c_i \mid X_1, X_2, X_3\right) \propto P\left(X_1 \mid c_i\right) P\left(X_2 \mid c_i\right) P\left(X_3 \mid c_i\right) P\left(c_i\right) \tag{2.5}$$

After the NB probabilities calculations, the algorithm selects the final class by using the Maximum A Posteriori decision rule, as explained above. Naive Bayes classifier will be used in the proposed Cybercrime Classification System of this thesis due to its simplicity and efficiency.

Chapter 3

Overview of Cybercrime Incident Architecture

This thesis has its foundations based on the paper *A Cybercrime incident architecture with adaptive response policy* (Tsakalidis et al.) [1]. Hence, it is needed to present an overview of the aforementioned paper in order to understand the wider context of the Cybercrime domain and, eventually, combine the basic principles of the paper with this thesis by giving a more practical approach to its functions.

3.1 Introduction

The paper discusses the evolving nature of Cybercrime and states that, due to its complex nature, it is difficult to develop a complete system that manages to categorize and tackle Cybercrime Incidents, along with evaluating the severity and proposing the appropriate response measures. It highlights the usefulness of Cybercrime classification which includes monitoring and evaluation of Cybercrime occurrences.

By utilizing such a system, authorities and policy-makers can make their decisions regarding incident response decisively and systematically. It provides an approach for analyzing CIs and triggering an adaptive response. More specific, the Cybercrime Incident Architecture aims at:

- generating insight and patterns regarding CIs
- evaluating and monitoring threat severity of CIs

- producing actionable and appropriate response policies and guidelines for stakeholders

The paper proposes a Cybercrime Incident Architecture for handling and mitigating Cybercrime Incidents, which consists of four components:

- Component I: CI Features
- Component II: Classification & CI Schema
- Component III: Threat Severity
- Component IV: Adaptive Response Policy

This architecture is shown in the figure:

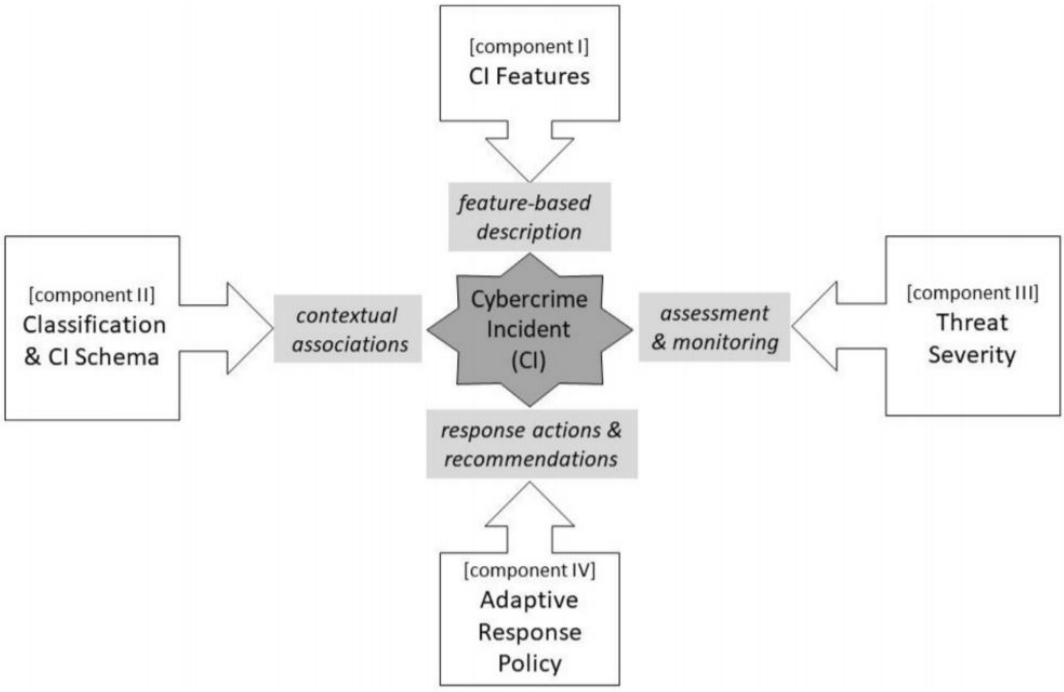


Figure 3.1: CIA Architecture. [1]

3.2 The components

3.2.1 Component I: CI Features

The purpose of Component I is to describe a CI in a systematic manner by identifying its distinctive features. These features describe the purpose of the CI and the subjects who suffered from it, along with the consequences of Cybercrime.

By properly examining this information, analysts can analyze the possible associations between them and gain valuable new knowledge from them, such as recurring patterns in Cybercrime.

Component I produces a feature-based description of the CI in the following format: *s: A case of [OCCURENCE] committed by the [OFFENDER(S)], conducted through [ACCESS VIOLATION], against the [TARGET] of [VICTIM(S)] results in [HARM].*

The paper quotes a table that shows the structured format of the CI Features produced from the Component:

Updated CI feature	Feature description
OCCURRENCE (INCIDENT)	Short factual description of the incident (e.g., *dissemination of phishing emails*)
OFFENDER	Individual or entity responsible for the incident (e.g., *cyber-criminal, online social hacker*)
ACCESS VIOLATION	Violation approach (e.g., *physical tampering, remote computer access*)
TARGET	Values that are the designated target (individual or collective)
VICTIM	Individual or entity that has suffered
HARM	Caused harm (e.g., *loss of property, emotional distress, moral harm*)

Figure 3.2: Component I: List of CI Features. [1]

3.2.2 Component II: Classification & CI Schema

The aim of Component II is to provide a comprehensive offense classification system for CIs. By classifying CIs and defining a relative framework, analysts can detect the possible associations and correlations between Cybercrime features and find useful patterns. It can also assist the automated process of suggesting response actions depending on the Cybercrime type and contextualizing the CI features.

The authors highlight the importance of classifying Cybercrime Incidents and state that it has been a challenge for researchers to produce a classification system which produces credible and accurate results due to Cybercrime's complex domain.

The paper suggests a 2-layer Classification System as below:

Layer 1	Layer 2
Type A Offences against the confidentiality, integrity and availability of computer data and systems	A1. Illegal Access (hacking, cracking) A2. Illegal data acquisition (data espionage) A3. Illegal Interception A4. Data Interference A5. System Interference A6. Misuse of devices
Type B Computer-related offences	B1. Computer-related forgery B2. Computer-related fraud B3. Identity theft
Type C Content-related Offences	C1. Pornographic Material C2. Child Pornography C3. Religious Offences C4. Cyberbullying C5. Illegal gambling and online games C6. Spam and related threats C7. Racism and hate speech on the Internet
Type D Offences related to infringements of copyright and related rights	D1. Copyright-related offences D2. Trademark-related offences
Type E Combinational offences	E1. Phishing E2. Cyber laundering E3. Cyberwarfare E4. Terrorist use of the Internet

Figure 3.3: Component II: Offence classification layers. [1]

3.2.3 Component III: Threat Severity

Component III aims to assess a CI based on its past occurrences and their perceived severity. It focuses on cybersecurity threat frequency and on monitoring Cybercrime offenses.

As stated in the paper, a severity assessment method for Cybercrime with respect to their frequency and the proposal of appropriate response measures is absent from the literature. Therefore, the proposed architecture, and especially Components III & IV, attempts to solve this problem.

Specifically, Component III:

- provides a formal method for qualitative analysis of cybersecurity threats

- evaluates their threat severity based on time progression

Component III classifies Cybersecurity threats in three categories, in relation to the past year:

- plus (+): indicates a rise in the frequency of a threat
- minus (-): indicates a drop in the frequency of a threat
- zero (0): indicates frequency steadiness

In addition, the component labels a threat based on its severity :

- Hyper-critical Threat (HCT)
- Critical Threat (CT)
- Active Threat (AT)
- Neutral Threat (NT)
- Diminishing Threat (DT)

In conclusion, Component III manages to incorporate active monitoring of cyber threats by detecting the frequency of each CI and by labeling a cyber threat based on its past and current trends. It can also assist in applying the appropriate response measures and also assign CIs to the relevant stakeholders, who are responsible for mitigating the said cyber threat.

For example, below are presented the trends of cyber threats: Spam, Insider Threat & Identity Theft:

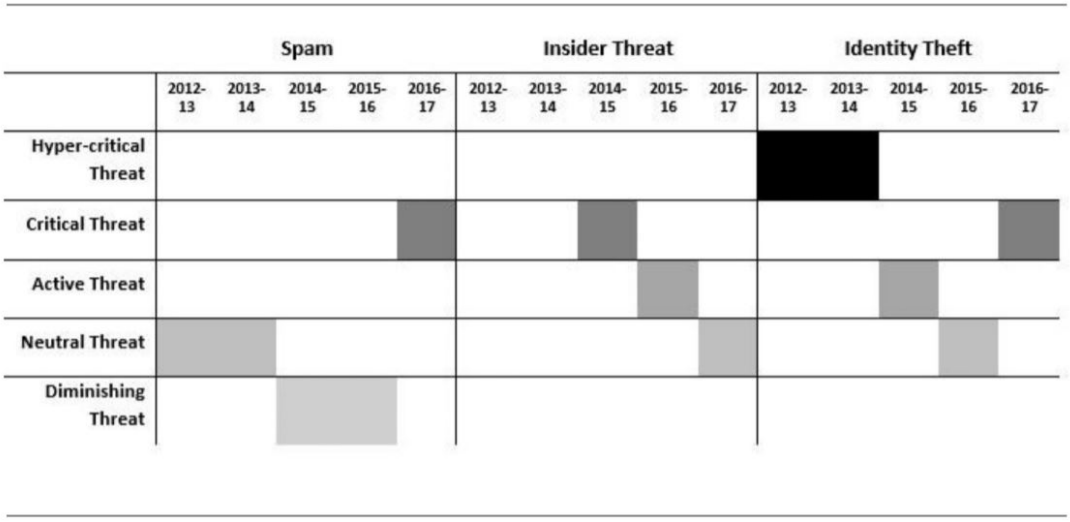


Figure 3.4: Component III: Threat Severity for specific CIs. [1]

3.2.4 Component IV: Adaptive Response Policy

Component IV aims to design an Adaptive Response Policy (ARP) which will produce:

- immediate actions to handle CIs
- specific measures to prevent similar CIs
- policies for a specific type of Cybercrime

The paper highlights that it is critical to take specific actions, measures and policies in order to fully confront a CI. Component II, which is about Classification, plays a significant role in the systematic operation of Component IV.

The authors present three steps for designing the ARP framework:

1. Define a list of stakeholders, derived by the query: ”After a CI occurrence, who is responsible for responding and at what level?”
2. Define a list of response measures and tackling actions, derived by the query: ”What happens during or immediately after a CI occurrence?”
3. Define a list of preventive measures and policies, derived by the query: ”What can be done to prevent similar CIs from reoccurring?”

The below figure gives an overview of the ARP framework in relation to incident occurrence:

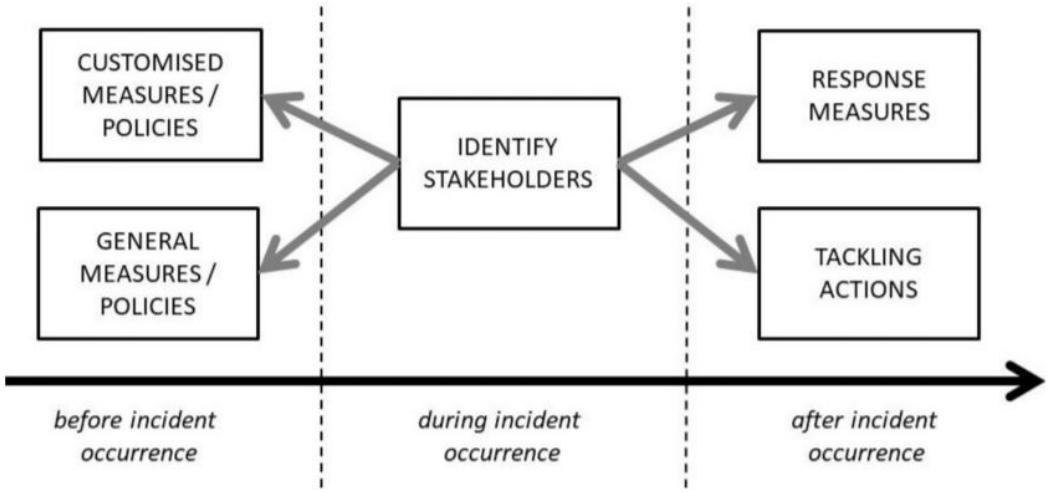


Figure 3.5: [1]

We can understand that *before incident occurrence* it is crucial to 1) apply customized measures/policies and 2) apply general measures/policies. *During incident occurrence*, Component IV attempts to identify and update the corresponding stakeholders. Finally, *after the incident occurrence*, the ARP framework proposes the proper response measures and tackling actions in order to successfully confront the CI.

3.3 Conclusions

The *Cybercrime incident architecture with adaptive response policy* proposed by *Tsakalidis et al.* [1], manages to provide a comprehensive framework for preventing, analyzing and tackling Cybercrimes with a systematic approach. It uses novel methods such as an Adaptive Response Framework and provides solid theoretical foundations which can be further expanded in order to approach the general problem presented in the paper in a more practical way.

The Cybercrime Classification System of this thesis proposes a practical approach with respect to *Components I & II* of the paper [1] in order to solve the Cybercrime Classification problem and to contribute to the actual implementation of the theoretical architecture.

Chapter 4

Cybercrime Classification System (CCS)

4.1 Introduction

In this Chapter, we will propose a **Cybercrime Classification System (CCS)** that attempts to present a practical approach for the Classification problem, based on the conceptual paper *A Cybercrime incident architecture with adaptive response policy* (Tsakalidis et al.)[1]. The CCS focuses on Component 1 and Component 2 of the aforementioned paper [1] and expands theirs theoretical functionalities by using Data Mining techniques such as Text Mining and Classification.

The purpose of the CCS is to classify a **Cybercrime Incident (CI)** based on its features (attacker, target, etc.) into a predefined category of Cybercrime (e.g. financial fraud, child pornography, phishing). This procedure can be expressed as a function: $y = M(x)$, where:

- $y \in c_1, c_2, ..., c_k$ and each c_i is a Cybercrime class,
- x is a vector that contains the features of a Cybercrime incident,
- M is the classifier model.

The automation of such a process is particularly useful for police authorities who collect such data for analysis. Specialized data analysts, with the proper interpretation of the findings, can make a decisive contribution to the prevention of Cybercrime, to the prediction of Cybercrime trends and finally propose various response measures.

One issue is that when a Cybercrime is first recorded and documented by the authorities, data does not have a specific structure but instead it is usually unstructured and expressed in free text. The proposed system addresses this issue by using text mining techniques and by transforming free text into structured data. In particular, it turns data into a vector that represents a Cybercrime incident with its features. This transformation is necessary for the Cybercrime Classification System to work properly.

The proposed CCS consists of three components:

- Component 1: Synthetic Data Generation
- Component 2: Evaluation & Data Preprocessing
- Component 3: Training & Classification-Prediction

Each component is composed of smaller processes. It is mandatory that these processes are successfully completed in order for the CCS to work properly. The detailed structure of each component is presented afterwards.

Also, below are presented the Symbols used in this thesis and their explanations, for the sake of brevity:

Abbreviation	Definition
CCS	Cybercrime Classification System
CC	Cybercrime Class
CI	Cybercrime Incident
CI vector	One-dimensional vector composed of the CI's 5 features

Table 4.1: Symbol notation

Prerequisites

For the data representation, it is needed to map Cybercrime features to numbers. The structure of each Cybercrime Incident (including its features) is derived from the paper *A Cybercrime incident architecture with adaptive response policy* (Tsakalidis et al.) [1]. This mapping is important due to the fact that, in general, classification algorithms tend to run better and more efficiently with numbers rather than text. This transformation is about **Feature Extraction**. In particular, the values of these features need to be mapped:

1. Offender

2. Access violation
3. Target
4. Victim
5. Target
6. Cybercrime class

It is obvious that the Cybercrime class is not known in advance. The other 5 features are known a priori and will be the ones that will determine, in collaboration with the classification algorithm, the final Cybercrime class to which a Cybercrime Incident falls. The explanation of these features is presented in the Background chapter.

The numerical mapping of the six above features is presented in the following tables (all the attribute values are also derived from paper *A Cybercrime incident architecture with adaptive response policy* (Tsakalidis et al.) [1]):

OF_1	**Abusive user**
OF_2	**Cyber-bully**
OF_3	**Cyber-criminal**
OF_4	**Cyber-fighter**
OF_5	**Cyber-terrorist**
OF_6	**Hacktivist**
OF_7	**Insider (employee)**
OF_8	**Online social hacker**
OF_9	**Script kiddie**
OF_{10}	**Sexually deviant user**
OF_{11}	**Company/Organization**

Table 4.2: Numerical mapping: offender

AV_1	**Physical tampering**
AV_2	**Local access**
AV_3	**Remote access**

Table 4.3: Numerical mapping: access violation

T_1	**Information and Communication Technologies (ICT)**
T_2	**ICT Infrastructure**
T_3	**Social**
T_4	**Financial**
T_5	**Emotional**

Table 4.4: Numerical mapping: target

V_1	**Individual**
V_2	**Company/Organization**
V_3	**Country/State**

Table 4.5: Numerical mapping: victim

H_1	**Infrastructure**
H_2	**Individual**
H_3	**Social**
H_4	**Inchoate**

Table 4.6: Numerical mapping: harm

CC_1	**Offences against the confidentiality, integrity and availability of computer data and systems**
CC_2	**Computer-related offences**
CC_3	**Content-related Offences**
CC_4	**Offences related to infringements of copyright and related rights**
CC_5	**Combinational offences**

Table 4.7: Numerical mapping: Cybercrime class

4.2 CCS architecture and components

The construction and operation of a classification system includes two general, distinct phases:

— Phase 1: Training

— Phase 2: Classification

It should be noted that these two phases exist in every classification system and should not be confused with the components of the proposed system, which will be presented in detail below. Also, it is not necessary that the components of the proposed system are executed in the order in which they are presented; another flow is required for training and another for prediction.

The architecture of the CCS is the following:

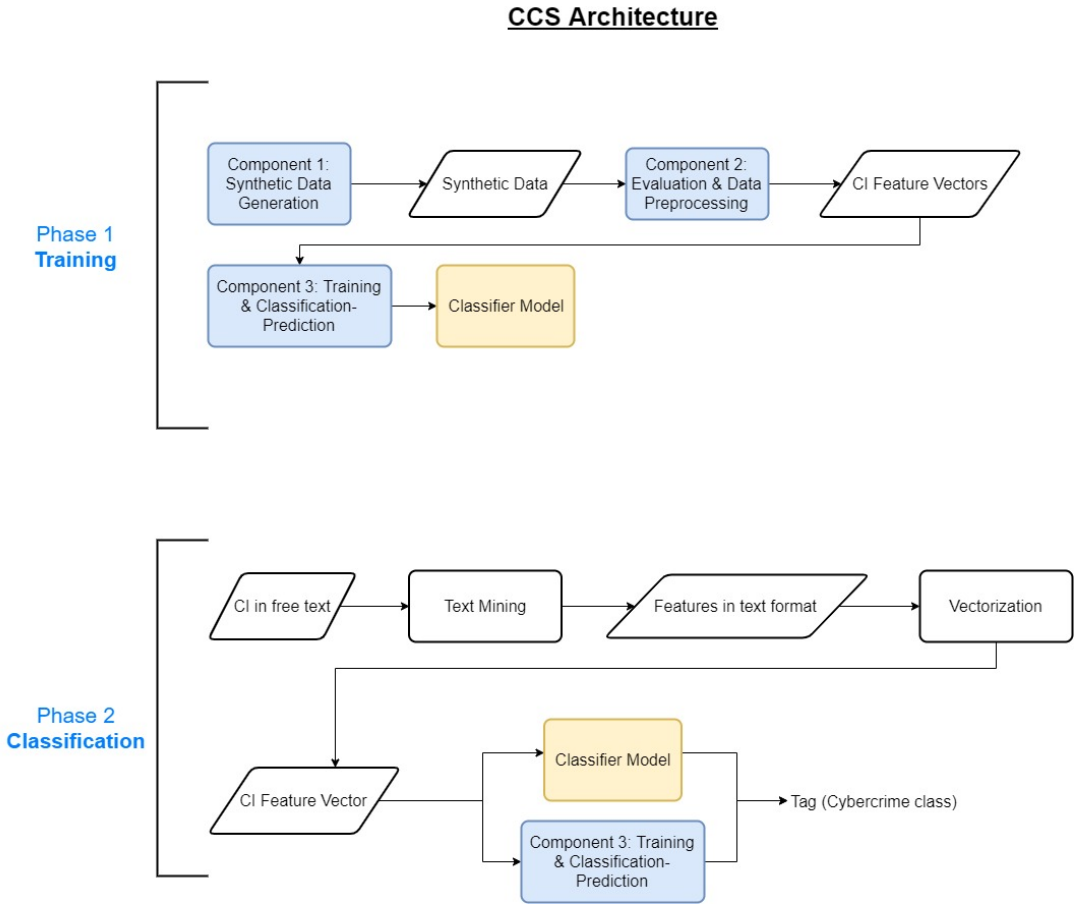


Figure 4.1: CCS architecture.

Although CCS architecture will be analyzed in detail later, we are going to give a brief explanation of the figure 4.1:

— Phase 1: Training

- *Component 1: Synthetic Data Generation*: During this step, the CCS will generate a synthetic dataset to use for training the model. The method for generating the data is by using a script that produces all the possible combinations of CIs, based on the correlations between CI features and Cybercrime Classes as shown in tables 4.9 to 4.13.

- *Synthetic Data*: Represents the produced dataset from *Component 1*.

- *Component 2: Evaluation & Data Preprocessing*: During this step, the CCS will evaluate the performance of Classification algorithms on the produces dataset in order to choose the most efficient. Afterwards, it will preprocess the dataset in order to improve its quality and make it work properly.

- *CI Feature Vectors*: The Feature Vectors are now the produced dataset but processed, cleaned and ready to train and classify.
- *Component 3: Training & Classification-Prediction*: Component 3, during the training phase, produces the classifier model based on the selected algorithm and on the input dataset. This step is very crucial for the CCS since Prediction's success depends on the classifier model.
- *Classifier Model*: The output of Component 3. The Classifier Model is the core of the CCS.

— Phase 2: Classification

- *CI in free text*: At first, the authorities will capture the CI in the form of free text. They will describe the CI in a systematic manner, as described in the paper *A Cybercrime incident architecture with adaptive response policy* (Tsakalidis et al.)[1].
- *Text Mining*: In this step, a text mining process takes place. Specifically, using text mining techniques, it will convert the free text into discrete features in order for the CI to be able to be given as input to the Classifier.
- *Features in text format*: After text mining, the features are extracted but they are still in text format.
- *Vectorization*: Here, the CCS will use the numerical mapping tables as presented in tables 4.2 to 4.7 in order to convert them to a numerical representation.
- *CI Feature Vector*: It is the output from the *Vectorization* step. It expresses the CI in the desired form.
- *Classifier Model* along with *Component 3: Training & Classification-Prediction*: The CI Feature vector is now given as input to Component 3, which uses the Classifier Model in order to predict the CC in which the CI falls.
- *Tag (Cybercrime Class)*: The final predicted CC (output of the previous step).

4.2.1 Comparing and selecting the proper Classification algorithm

In this section, we will compare 4 of the most used Machine Learning Classification algorithms: *kNN (k-Nearest-Neighbors), Decision Trees, Support-vector machine* and *Naive Bayes*, and finally we will choose the best fit for our Classification problem.

Choosing the right ML algorithm for a Classification problem is often very challenging. There is not an algorithm that fits to all Classification problems; we need to carefully **examine which algorithm is the best fit for our Classification problem with respect to accuracy and ease of use**. The selection also depends on many factors from the type of problem.

Overview of the algorithms

In this section, a general overview of each algorithm will be presented in order to get a fairly good understanding of how they work and to be able to compare them in detail in the next paragraphs.

1. kNN (k-Nearest-Neighbors) — kNN is a non-parametric algorithm (=an algorithm which does not make assumptions about the form of the model) which assigns the class label of the nearest set of previously labeled points [40]. It is a ”lazy” method which means that it does not require training. In general, it achieves lower accuracy since there is no standard/optimal way to choose k value. It is sensitive to noise and its performance is analogous to the size of the dataset since every point must be revisited in order to calculate the dominant class of the nearest neighbors.

2. Decision Trees — Decision Trees include algorithms such as ID3, C4.5 (J48), C5.0 and others. They are fairly simple to understand and describe, and they can manage feature interactions with ease and efficiency [41]. Decision Trees algorithms are based on a variety of splitting criteria (GINI coefficient, Support, Info Gain and others). It is worth noting that although Decision Trees need a relatively small amount of time to build their models, it takes a lot of time to run the Classification because of the tree building process. DTs are not appropriate for problems with a large number of classes and big datasets [42], but they are useful when a Classification problem has a low domain complexity and a medium-sized dataset.

3. Support-vector machine — SVM is an algorithm that generally achieves high accuracy but is quite complex in terms of tuning. It works by finding the hyperplane which differentiates the classes the most. SVM algorithms are efficient with high dimensional data and performance, unlike kNN, is based on the number of training cycles instead of the number of sample points [41].

4. Naive Bayes — The Naive Bayes Classifier represents a Bayesian Network with only one parent and many children. It uses conditional probabilities as its core process. In terms of speed, training time is quite fast and due to the fact that it does not require any parameter to set, it is also fast to configure. Naive Bayes is not advised to use with high dimensional data.

Ease of use

In this section, we will examine the ease of use and the simplicity of each Classification algorithm. As a general rule, if the underlying algorithm is highly complex and if it demands a large number of parameters to be set, it is logical to conclude that it does not belong to easy-to-use algorithms.

1. kNN (k-Nearest-Neighbors) — kNN is a simple algorithm and it only requires setting k value. k refers to the number of neighbors which kNN must visit in order to classify a sample point. There is not an optimal way to select k value, so it is advised to set and test repeatedly until sufficient accuracy is achieved. So, although we can conclude that kNN is easy to use, we should take into consideration that the selection of k value is critical and does slow down the process of tuning the Classification algorithm.

2. Decision Trees — Decision Trees generally require many parameters to be set. Different metrics-parameters such as GINI or Entropy, max depth, number of splits and others, make DTs quite complex to get them tuned. One must have a very good understanding of the underlying algorithm, otherwise it is challenging to correctly set all the parameters.

3. Support-vector machine — SVM is one of the most complex classification algorithms and therefore it is logical to involve tuning of parameters to work, such as

C, $Gamma$, $Kernel$. Since performance is heavily affected by the selection of parameters, SVM is not recommended for not so complex Classification problems with a medium-sized dataset.

4. Naive Bayes — The Naive Bayes Classifier is easy to implement and quick to run. It does not require any tuning, since it is based on very specific, pre-determined rules (an expansion of the Bayes' Theorem). Despite its simplicity, Naive Bayes often manages to outperform much more sophisticated classifiers.

Accuracy Comparison

Performing an experiment is useful when trying to choose the proper Classification algorithm. Although the fundamentals should also be taken into consideration, the actual performance of each algorithm is going to reveal whether an algorithm is a good fit for our Classification problem.

For this experiment, we will test each algorithm on the training dataset itself and compare them based on two important metrics:

1. Classification accuracy
2. Recall

Classification accuracy is the percentage of the correctly classified instances, while **Recall** is the measure of our model correctly identifying True Positives. Higher is better in both metrics. The formulas are accordingly:

$$Classification\ Accuracy = \frac{Correctly\ Classified\ Instances}{Total\ Instances}$$

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)}$$

Below are the results of the experiment:

Algorithm	Accuracy (%)	Recall (weighted avg.)
kNN (k=2)	93,33	0,933
Decision Trees (J48)	92,8	0,928
Support-vector machine	88	0,88
Naive Bayes	93,6	0,936

Table 4.8: Algorithms comparison while testing on training set.

Naive Bayes has the higher Accuracy and the higher Recall and therefore it is the best fit for our Classification problem in terms of performance.

Verdict

Considering all the above, the CCS will use the **Naive Bayes Classifier** due to its simplicity and efficiency. The experimental comparison also showed that NB is a good fit for our Classification problem. Naive Bayes is also considered a reliable algorithm since it's been used widely with great results.

4.2.2 Phase 1: Model Training

The proposed CCS consists of 2 distinct phases: *Training* and *Classification-Prediction*. It is important to understand how these phases differ and how they work under the hood before we proceed to the actual Classification System.

Classification using machine learning, instead of being based on "handmade" rules, learns to predict/classify based on previous observations. Using already categorized examples as a learning dataset, the classification algorithm learns the correlations between the data and can predict that specific input data corresponds to specific tag classes. A necessary procedure for this technique is **Model Training**.

At first, we need to **extract features** from the data (Feature Extraction) and to transform the raw data into discrete -usually numerical- representations. In the case of the proposed system, the raw data will be converted into vectors. This process is called **vectorization**.

The machine learning algorithm then receives the training dataset which already has the class tag for each Cybercrime Incident. Thus, the algorithm produces the Classification model.

The above procedure is summarized in the following figure:

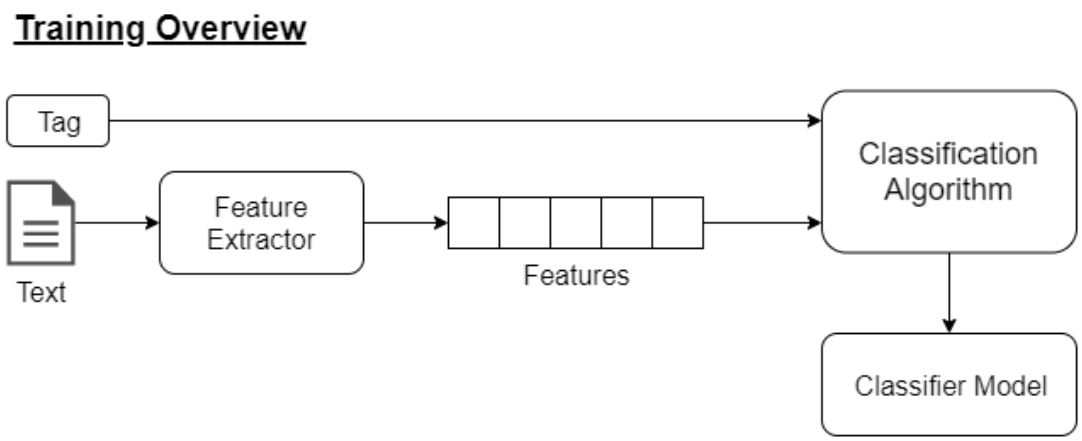


Figure 4.2: Training overview.

4.2.3 Phase 2: Classification - Prediction

After completing the training phase with enough data, the model can now make predictions with relatively sufficient accuracy. Prediction is essentially the execution of the classifier model.

As shown in the figures, until the features are extracted, the prediction phase is the same as the training phase. However, then, the vectors are given as input to the classification model and provide the class label to which the specific Cybercrime Incident belongs.

Success rates of the prediction depend directly on the quality of the learning set. If the model is improperly trained, the model is expected to have low success rates and therefore reduce its effectiveness. The Prediction process is shown in the following figure:

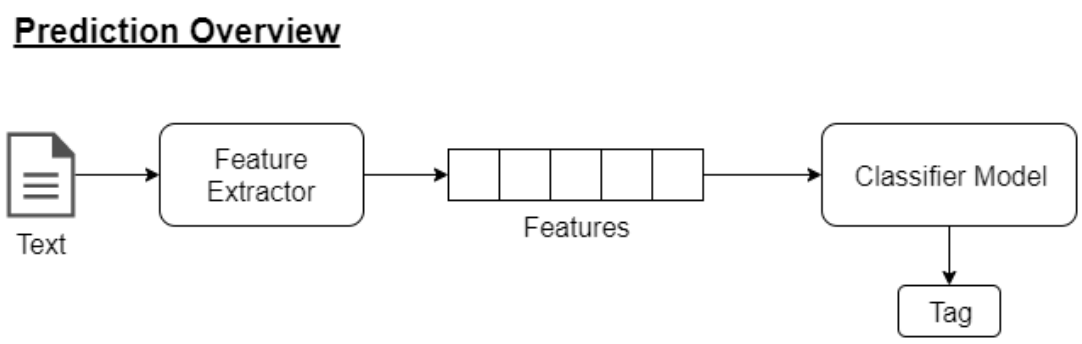


Figure 4.3: Prediction overview.

4.2.4 Component 1: Synthetic Data Generation

In order to evaluate our classification model, we need to perform some experiments. To do so, it is needed to generate synthetic data which will be given to our model for the Training phase. Component 1 is responsible for this process and below we will present its structure and how it works.

Correlations between Cybercrime classes and Cybercrime features

In order to proceed with the Synthetic Data generation, it is required to define the correlations between Cybercrime classes and Cybercrime features. In specific, it is needed to research and conclude which Cybercrime features fall into each Cybercrime class so we can **generate all the possible combinations applicable to each class**. For example, let's assume that a CI belongs to CC_1 (see 4.7); not all features can logically belong to this class. CC_1 is about *Offences against the confidentiality, integrity and availability of computer data and systems* and therefore it is irrational for this CI to have features such as T_5 (see 4.4) which refers to emotional target. These correlations are presented, for each Cybercrime class, in the below tables:

CC_1						Count
Offender (OF_i)	OF_1	OF_3	OF_6	OF_7	OF_9	5
Access Violation (AV_i)	AV_1	AV_2	AV_3			3
Target (T_i)	T_1	T_2	T_4			3
Victim (V_i)	V_1	V_2	V_3			3
Harm (H_i)	H_1	H_2				2
Permutations: 5*3*3*3*2 = 270						

Table 4.9: Applicable feature values for CC_1

CC_2				Count
Offender (OF_i)	OF_3	OF_8		2
Access Violation (AV_i)	AV_3			1
Target (T_i)	T_3	T_4	T_5	3
Victim (V_i)	V_1			1
Harm (H_i)	H_2	H_3		2
Permutations: 2*1*3*1*2 = 12				

Table 4.10: Applicable feature values for CC_2

CC_3						Count
Offender (OF_i)	OF_2	OF_4	OF_5	OF_8	OF_10	5
Access Violation (AV_i)	AV_3					1
Target (T_i)	T_3	T_5				2
Victim (V_i)	V_1					1
Harm (H_i)	H_2	H_3	H_4			3
Permutations: 5*1*2*1*3 = 30						

Table 4.11: Applicable feature values for CC_3

CC_4				Count
Offender (OF_i)	OF_1	OF_3	OF_7	3
Access Violation (AV_i)	AV_3			1
Target (T_i)	T_4			1
Victim (V_i)	V_2			1
Harm (H_i)	H_2			1
Permutations: 3*1*1*1*1 = 3				

Table 4.12: Applicable feature values for CC_4

CC_5						Count
Offender (OF_i)	OF_1	OF_3	OF_4	OF_5	OF_{11}	5
Access Violation (AV_i)	AV_3					1
Target (T_i)	T_3	T_5				2
Victim (V_i)	V_1	V_3				2
Harm (H_i)	H_2	H_3	H_4			3
Permutations: 5*1*2*2*3 = 60						

Table 4.13: Applicable feature values for CC_5

Synthetic Data Definition

Synthetic data is a set of data that is generated programmatically, usually with scripts. The purpose of generating Synthetic Data is to create a useful and robust dataset for training a machine learning model, in cases where real-world data are not enough or do not exist.

Synthetic data are necessary in our case, since there are no publicly available datasets regarding Cybercrimes, due to safety and privacy reasons. Therefore, with the proper techniques, we can create data that meet the requirements and also adjust them to real-world incidents. It is, in general, a quite fast way to acquire data since it does not require searching extensively for a specific form of data nor heavy preprocessing, such as removing noise and outliers, duplicates and false data entries.

Synthetic Data Generation script & methodology

As explained in previous chapters, each Cybercrime class has 5 subsets; one for each CI feature. In order to generate the synthetic data, we need a programming script which loops in these subsets for each class and, finally, produces all the possible Cybercrime Incidents in the form of a CI vector. These CI vectors, all combined, are going to compose the final dataset. The logic behind it is fairly simple and is presented below in the form of pseudocode:

```
Function GenerateSyntheticData()

For class = 1 to 5 //For each Cybercrime class
    offenders[] = getOffenders(class)  //Returns offenders applicable to each class
    accessViolations[] = getAccessViolations(class) //Returns access violations applicable to each class
    targets[] = getTargets(class) //Returns targets applicable to each class
    victims[] = getVictims(class) //Returns victims applicable to each class
    harms[] = getHarms(class) //Returns harms applicable to each class

    For offender in offenders
        For accessviolation in accessviolations
            For target in targets
                For victim in victims
                    For harm in harms
                        WriteLineToCSV(offender, accessviolation, target, victim, harm, class)

Endfunction
```

Figure 4.4: Pseudocode for Synthetic Data generation.

The produced *.csv* file has the below format:

```
offender,accessviolation,target,victim,harm,ccClass
3,3,3,1,2,2
3,3,3,1,3,2
3,3,4,1,2,2
3,3,4,1,3,2
3,3,5,1,2,2
3,3,5,1,3,2
8,3,3,1,2,2
8,3,3,1,3,2
8,3,4,1,2,2
```

Figure 4.5: A sample of the final .csv file.

4.2.5 Component 2: Evaluation & Data Preprocessing

Component 2 is about Evaluating and Preprocessing our dataset. During the Evaluation process, the CCS will analyze and evaluate the generated data in order to conclude whether the dataset is qualitative or not. Then, the CCS will apply some Preprocess-

ing filters which are going to further improve the dataset so we can produce a more efficient Classification model.

Evaluating the generated data — The generated dataset now contains all the possible combinations of Cybercrime Incidents according to the above tables which define the correlations between features and Cybercrime Classes.

By importing our dataset into Weka we can visualize the distribution of instances for each class:

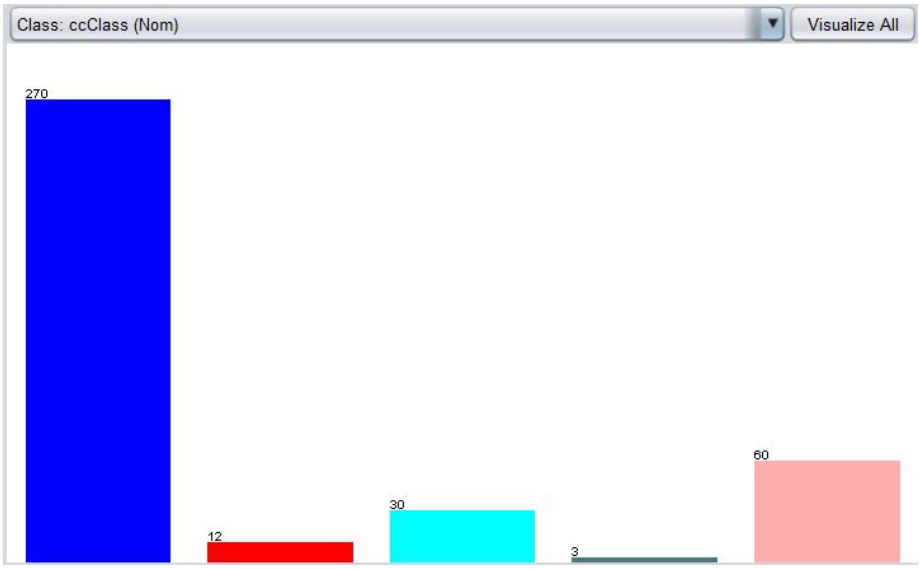


Figure 4.6: Distribution of instances for each class.

- CC_1: 270 instances
- CC_2: 12 instances
- CC_3: 30 instances
- CC_4: 3 instances
- CC_5: 60 instances

Because of the nature of our domain (Cybercrime), we can observe that the dataset has **imbalanced classes**. When a multiclass classification problem has significantly less data in one class than in the others, it tends to ignore the minority class(es) and classify most instances into the majority class [43]. This is called an **imbalanced classification problem**. This happens due to the fact that Naive Bayes classifier takes into

consideration the prior probabilities of $P(CC_i)$. It is obvious that this model is not useful, nor accurate, and thus it needs improvement. These are the class weights, before any preprocessing:

Class (CC_i)	Number of instances	Weight
1	270	270.0
2	12	12.0
3	30	30.0
4	3	3.0
5	60	60.0

Table 4.14: Weights of the classes before reweighting (class balancing)

Testing — Now, let's test the accuracy of our classification model before any preprocessing for a first, quick evaluation. Note that the process and the details of the classification algorithm will be analyzed later. Testing our model with the training dataset itself, with a cross-validation of 10 folds, shows a classification accuracy of 91,73%:

```
=== Summary ===

Correctly Classified Instances         344               91.7333 %
Incorrectly Classified Instances        31                8.2667 %
Kappa statistic                          0.8113
Mean absolute error                      0.0387
Root mean squared error                  0.1365
Relative absolute error                 21.3014 %
Root relative squared error             45.5586 %
Total Number of Instances              375
```

This means that our model classified correctly the 91,73% of all instances, namely 344 out of 375 instances. We can conclude that our model has a quite good training dataset and that the algorithm fits to our classification problem, but this does not mean that this accuracy will apply to every test set.

By looking at the Confusion Matrix, we understand that class 4 (CC_4) has all of its instances classified incorrectly (3 out of 3). This happens mainly due to the fact that we described before (imbalanced classes).

```
=== Confusion Matrix ===

   a   b   c   d   e   <-- classified as
 270   0   0   0   0 |   a = 1
   1   3   4   0   4 |   b = 2
   0   0  18   0  12 |   c = 3
   3   0   0   0   0 |   d = 4
   0   0   7   0  53 |   e = 5
```

Now, let's test our model with a supplied test set which contains the below instances:

- CI_1: 3,1,2,3,2,?
- CI_2: 8,3,4,1,2,?
- CI_3: 10,3,3,1,3,?
- CI_4: 3,3,4,2,2,?
- CI_5: 11,3,3,1,2,?

The "?" means that the last feature is the target class and is unknown to Weka. The test set has 5 instances advisedly; first instance belongs to CC_1, the second instance belongs to CC_2 and so on. Note that Weka is not aware of that but it helps the analysts to correctly interpret the classification results.

These are the results:

```
=== Predictions on test set ===

 inst#     actual  predicted error prediction
     1        1:?        1:1       0.998
     2        1:?        2:2       0.842
     3        1:?        3:3       0.853
     4        1:?        1:1       0.904
     5        1:?        5:5       0.842
```

We can see, once again, that the 4th instance (CI_4 which we know it belongs to CC_4) has been classified incorrectly to CC_1. The other 4 instances have been classified

correctly and this is a good sign that our classification model works accurately. In the next section, we will try to solve this confusion by using a Class Balancing filter.

Data Preprocessing — Data preprocessing refers to a data mining technique which is useful for transforming data in a more useful and efficient format. Data preprocessing involves various subprocesses, such as data cleaning, data transformation and data reduction [44].

Data Cleaning: Data Cleaning is a process that attempts to solve problems related mainly with duplicate instances and missing values [45]. ”Dirty” data, noise and outliers can make the Classification model considerably less reliable [44] and therefore it is important to try to address such issues.

Because in our Classification problem the dataset is synthetically generated, we do not face such problems since we have methodically and systematically created the training dataset so it does not contain duplicates instances nor missing values (see 4.2.4).

Class Balancing: The previous experiments made it obvious that the classes in our Classification problem are indeed **imbalanced**. For instance, CC_1 has a weight of 270, whereas CC_4 has a weight of 3 and this is the reason why the experiment did not achieve the correct classification for CC_4.

As we described before, when a multiclass classification problem has significantly fewer data in one class than in the others, it tends to **ignore the minority class(es) and classify all instances into the majority class** [43]. In order to solve this issue, Weka provides us with a useful filter called **Class Balancer**. Class Balancer sets **equal weights for each class** so that the classifier does not have a preference towards the class with the most instances. So, instead of duplicating instances of a minority class in order to make all classes even, we can apply this filter which basically has the same effect according to Weka documentation [46]. In terms of programming, the Class Balancer preprocessing filter implements Weka’s WeightedInstancesHandler which is a way that Weka assigns weight to each instance and therefore makes it possible to balance classes. The count for each class remains the same since not instances are added or removed from the dataset. After reweighting, the class weights are as below:

Class (CC_i)	Number of instances	Weight
1	270	75.0
2	12	75.0
3	30	75.0
4	3	75.0
5	60	75.0

Table 4.15: Weights of the classes after reweighting (class balancing)

Reweighting formula: $W = \frac{x_i + ... + x_n}{c} = \frac{375}{5} = 75$

- $W =$ weight of each class
- $C =$ number of classes in data set
- $x_i =$ number of instances of each class

Now, let's run NB Classifier again with the training set as a test set, as we did before reweighting:

```
=== Summary ===

Correctly Classified Instances         337.7778          90.0741 %
Incorrectly Classified Instances        37.2222           9.9259 %
Kappa statistic                          0.8759
Mean absolute error                      0.0622
Root mean squared error                  0.1639
Relative absolute error                 19.4258 %
Root relative squared error             40.964  %
Total Number of Instances              375


=== Confusion Matrix ===

   a     b     c     d     e      <-- classified as
 67.78  0.28  0     6.94  0     |     a = 1
  0    75     0     0     0     |     b = 2
  0    10    65     0     0     |     c = 3
  0     0     0    75     0     |     d = 4
  0     5    15     0    55     |     e = 5
```

We can see that the accuracy has slightly decreased when testing on the training set itself. However, in order to come to a safer conclusion, we must also test our classifier with the same test set as before:

```
=== Predictions on test set ===

    inst#     actual  predicted error prediction
        1        1:?        1:1       0.998
        2        1:?        2:2       0.984
        3        1:?        3:3       0.914
        4        1:?        4:4       0.984
        5        1:?        5:5       0.802

=== Evaluation on test set ===
```

This time, the 4th instance (CI_4) has been classified correctly to CC_4. Because CC_4 is a minority class, before reweighting (or Class Balancing) our classifier did not manage to correctly classify CIs which belong to CC_4. It is now obvious that class balancing is a critical process which, in most cases, must be done for a more accurate classification model [47].

4.2.6 Component 3: Training & Classification-Prediction

Training

Component 3 is about executing the Naive Bayes algorithm. It is responsible for both Training & Classification-Prediction, since these two processes are closely related, but it is important not to confuse those two processes. The Component 3 is an adaptive component meaning that it executes the proper process (Training or Classification) depending on the corresponding CCS phase, as explained in the previous sections.

If the CCS needs to use Component 3 for the Training phase, it receives the training dataset as an input from Component 2 (which has preprocessed the dataset) and builds the classifier model by using the Naive Bayes classifier. This process is handled by the Weka software which does all the calculations required for the NB classifier. Specifically, Weka calculates all the prior probabilities like we explained in 2.2.3 and then, based on these observations, it creates the Classifier Model.

If Component 3 needs to be used for the Classification-Prediction phase, it calculates the probabilities of the test data, based on the Naive Bayes classifier, and does the required comparisons with the training set in order to classify the new instances.

Text mining & vectorization

In order to classify a new instance, the CCS must vectorize the Cybercrime Incident which initially is in the form of free text. Therefore, text mining techniques are needed in order to do feature extraction. However, details of text mining are outside of the scope of this thesis so we will only give an overview of the process.

The raw data, in the context of the proposed system, is a brief description of the Cybercrime Incident in free text. **Feature extraction** will be done through text mining techniques, i.e. the system will vectorize the data to following format: $CI = (OF, AV, T, V, H, CC)$.

It is obvious that the position of the values in the vector determines the feature to which it refers (see numerical mapping). If we assume that we are in the training phase and therefore the data has a class tag, the last position of the vector related to the Cybercrime class will be $\neq 0$.

Assuming that we are in the classification phase, the input data has no value for the Cybercrime class (0), since this is the final target of the proposed system and it does not know it in advance.

For instance, assume that the Text Mining process produces this vector as result: CI = (Abusive User, Remote access, Financial, Company/Organization, Individual, 0). Then, using the numerical mapping tables (see tables 4.2 to 4.7), the vector will be converted to a numeric format: $CI = (1, 3, 4, 2, 2, 0)$. These vectors are equivalent and describe the same Cybercrime incident. Each number corresponds to a specific value of the feature, as already described ($1 \rightarrow$Abusive user, $3 \rightarrow$ remote access etc.). Note that the CC feature equals 0 assuming we are in the testing phase. The data transformation is also performed during model training, with the only difference that the input data already contains the CC of the CI.

Classification-Prediction

After the vectorization process is finished, we now have the CI incident in the desired format: $CI = (1, 3, 4, 2, 2, 0)$. The Classification process receives the CI as input and then proceeds with the class prediction using the Classifier Model which has been produced by Weka during the Training process.

Classification is the final process of the proposed Cybercrime Classification Sys-

tem since its final purpose is to classify a Cybercrime Incident to its corresponding Cybercrime Class. By automatically finding the Cybercrime Class of an incident, authorities and analysts can take advantage of this information in order to prevent, confront or predict a Cybercrime.

In the next chapter, we will present how the Cybercrime Classification System exactly works in real-life circumstances by focusing on a specific Case Study.

Example

It is meaningful to understand how the Naive Bayes algorithm actually classifies an instance. The proposed CCS makes use of Weka software for the Classification but in order to observe how the underlying process works, we will execute the Naive Bayes algorithm by hand with the following Cybercrime Incident: CI_x : $OF_{11}, AV_3, T_3, V_1, H_2, CC_x$. This CI actually belongs to CC_5 and this is where we want our Classifier model to classify this CI.

Firstly, we need to count how many times a feature value occurs in each CC in order to later calculate the required probabilities for Naive Bayes. This is called a **Frequency table**. For example, we need to know the percentage of instances in which the Offender is Company/Organization (OF_1) per Cybercrime Class. It is worth noting that Naive Bayes is apt to the **Zero Frequency Problem** which occurs when a feature value has zero occurrences in every class (zero frequency) [48] and therefore the frequency-based probability would be zero. Afterwards, this single probability will be multiplied with the other probabilities and therefore will set the whole probability equal to zero. An approach to solve this problem is to add one to the count for every feature value which has zero frequency. This process is called **Additive Smoothing** and Weka also uses this approach [48].

Weka has this information from the training dataset as shown below and has already smoothed the frequency table:

	CC_1	CC_2	CC_3	CC_4	CC_5
OF_1	55	1	1	2	13
OF_2	1	1	7	1	1
OF_3	55	7	1	2	13
OF_4	1	1	7	1	13
OF_5	1	1	7	1	13
OF_6	55	1	1	1	1
OF_7	55	1	1	2	1
OF_8	1	7	7	1	1
OF_9	55	1	1	1	1
OF_{10}	1	1	7	1	1
OF_{11}	1	1	1	1	13

Table 4.16: Offender (OF_i) frequency table

	CC_1	CC_2	CC_3	CC_4	CC_5
AV_1	91	1	1	1	1
AV_2	91	1	1	1	1
AV_3	91	13	31	4	61

Table 4.17: Access Violation (AV_i) frequency table

	CC_1	CC_2	CC_3	CC_4	CC_5
T_1	91	1	1	1	1
T_2	91	1	1	1	1
T_3	1	5	16	1	31
T_4	91	5	1	4	1
T_5	1	5	16	1	13

Table 4.18: Target (T_i) frequency table

	CC_1	CC_2	CC_3	CC_4	CC_5
V_1	91	13	31	1	31
V_2	91	1	1	4	1
V_3	91	1	1	1	31

Table 4.19: Victim (V_i) frequency table

	CC_1	CC_2	CC_3	CC_4	CC_5
H_1	136	1	1	1	1
H_2	136	7	11	4	21
H_3	1	7	11	1	21
H_4	1	1	11	1	21

Table 4.20: Harm (H_i) frequency table

Note again that the instance that we want to classify is:

$$CI_x : OF_{11}, AV_3, T_3, V_1, H_2, CC_x$$

- Calculate $P(CC_i) = \frac{CC_i \ instances}{Total \ Instances}$
  - $P(CC_1) = 270/375 = 0.72$
  - $P(CC_2) = 12/375 = 0.032$
  - $P(CC_3) = 30/375 = 0.08$
  - $P(CC_4) = 3/375 = 0.008$
  - $P(CC_5) = 60/375 = 0.16$

	CC_1	CC_2	CC_3	CC_4	CC_5
$P(CC_i)$	0.72	0.032	0.08	0.008	0.16

- Calculate $P(CI_x|CC_i)$. In this step, we will calculate the possibility of each of the five feature values of the CI_x to occur in each class.
  - Calculate $P(OF_{11}|CC_i)$

	CC_1	CC_2	CC_3	CC_4	CC_5
$P(OF_{11}\|CC_i)$	1/16	1/16	1/16	1/16	13/16

  - Calculate $P(AV_3|CC_i)$

	CC_1	CC_2	CC_3	CC_4	CC_5
$P(AV_3\|CC_i)$	91/200	13/200	31/200	4/200	61/200

  - Calculate $P(T_3|CC_i)$

	CC_1	CC_2	CC_3	CC_4	CC_5
$P(T_3\|CC_i)$	1/54	5/54	16/54	1/54	31/54

  - Calculate $P(V_1|CC_i)$

	CC_1	CC_2	CC_3	CC_4	CC_5
$P(T_3\|CC_i)$	91/167	13/167	31/167	1/167	31/167

– Calculate $P(H_2|CC_i)$

	CC_1	CC_2	CC_3	CC_4	CC_5
$P(H_2\|CC_i)$	136/179	7/179	11/179	4/179	21/179

- Finally, we will calculate:

  – $P(CI_x|CC_i) = P(CC_i) \cdot P(OF_{11}|CC_i) \cdot P(AV_3|CC_i) \cdot P(T_3|CC_i) \cdot P(V_1|CC_i) \cdot P(H_2|CC_i)$

  which will determine the target class. The target class will be selected based on the Maximum A Posteriori rule, as explained in the previous sections.

  – $\boldsymbol{P(CI_x|CC_1)} = P(CC_1) \cdot P(OF_{11}|CC_1) \cdot P(AV_3|CC_1) \cdot P(T_3|CC_1) \cdot P(V_1|CC_1) \cdot P(H_2|CC_1) = 0.72 \cdot 1/16 \cdot 91/200 \cdot 1/54 \cdot 91/167 \cdot 136/179 = \mathbf{0.015\%}$

  – $\boldsymbol{P(CI_x|CC_2)} = P(CC_2) \cdot P(OF_{11}|CC_2) \cdot P(AV_3|CC_2) \cdot P(T_3|CC_2) \cdot P(V_1|CC_2) \cdot P(H_2|CC_2) = 0.032 \cdot 1/16 \cdot 13/200 \cdot 5/54 \cdot 13/167 \cdot 7/179 = \mathbf{0.0000037\%}$

  – $\boldsymbol{P(CI_x|CC_3)} = P(CC_3) \cdot P(OF_{11}|CC_3) \cdot P(AV_3|CC_3) \cdot P(T_3|CC_3) \cdot P(V_1|CC_3) \cdot P(H_2|CC_3) = 0.08 \cdot 1/16 \cdot 31/200 \cdot 16/54 \cdot 31/167 \cdot 11/179 = \mathbf{0.00026\%}$

  – $\boldsymbol{P(CI_x|CC_4)} = P(CC_4) \cdot P(OF_{11}|CC_4) \cdot P(AV_3|CC_4) \cdot P(T_3|CC_4) \cdot P(V_1|CC_4) \cdot P(H_2|CC_4) = 0.72 \cdot 1/16 \cdot 4/200 \cdot 1/54 \cdot 1/167 \cdot 4/179 = \mathbf{2.48 \cdot 10^{-9}\%}$

  – $\boldsymbol{P(CI_x|CC_5)} = P(CC_5) \cdot P(OF_{11}|CC_5) \cdot P(AV_3|CC_5) \cdot P(T_3|CC_5) \cdot P(V_1|CC_5) \cdot P(H_2|CC_5) = 0.16 \cdot 13/16 \cdot 61/200 \cdot 31/54 \cdot 31/167 \cdot 21/179 = \mathbf{0.049\%}$

The highest probability $\boldsymbol{P(CI_x|CC_i)}$ is for $i = 5$ and specifically:

$$\boldsymbol{P(CI_x|CC_5) = 0.049\%} \tag{4.1}$$

. Based on the Maximum A Posteriori Rule, the Naive Bayes Classifier will classify the instance $CI_x : 11, 3, 3, 1, 2, ?$ to Cybercrime Class CC_5 which refers to *Combinational*

offenses. This Classification is **correct** since, as we stated before, we know that the aforementioned CI actually belongs to CC_5. The Cybercrime Classification System confirms this result since Weka also classifies the foresaid CI to CC_5.

4.3 Conclusions

This chapter presents a Cybercrime Classification System which analyzes the features of a Cybercrime Incident (victim, harm etc.) in order to classify new, unknown instances (CIs) using Machine Learning methods, such as Classification.

A reasonable question, after presenting the Classification System, is *why the Classification of CIs is useful*. In summary:

- it helps the authorities to propose the appropriate counter-measures in order to confront the CI
- it helps to recommend appropriate actions towards managing effective policies [49]
- it helps to monitor and handle similar CIs
- it helps to assess the threat severity of this Cybercrime type
- it helps to identify possible correlations between the features of the CI
- it assists the grouping of similar CIs
- it produces data that can be analyzed, e.g. find which Cybercrime class occurs more often, which type of offender commits the most severe Cybercrimes and others statistics.

Concluding, the Classification of Cybercrime Incidents is important to authorities and to relevant stakeholders because it allows them to quickly and automatically identify the type of Cybercrime and propose the appropriate counter-measures in order to confront the CI effectively. The produced data can also be analyzed for finding recurring patterns among CIs and attempt to prevent them from happening in the future.

Chapter 5

CCS in action: A case study

This chapter presents a case study that is about a Cybercrime Incident regarding possession and distribution of child pornography material. We will thoroughly explain the CI and then we will examine how the proposed Cybercrime Classification System handles the incident.

Firstly, we will describe the Cybercrime Incident in order to provide context and gain a good understanding of the occurrence. Then, we will demonstrate how the proposed CCS handles the CI in terms of classification and confrontation. Finally, we will highlight the importance of a Classification System and how it helps the authorities to mitigate similar incidents by analyzing relevant data.

5.1 The Cybercrime Incident

Author's Note: We should make clear that the Cybercrime Incident presented in this chapter does not correspond to a real-world incident and is an imaginary scenario of the author.

Cyberspace has become a very convenient place for finding and sharing illegal material. It provides all the tools needed: dark web, file storage, file sharing protocols (Peer-to-peer), anonymity, speed and untraceable payments using cryptocurrencies such as Bitcoin.

Child pornography cases have significantly increased in the last few years and heavily bother the authorities. In this Case Study, we will present a scenario that involves a person who downloads and shares (uploads) material which includes child

pornography.

The incident — Greek authorities have managed to gain access to a global illegal network that shares child pornography material via a peer-to-peer protocol and specifically Torrent. Users upload and download files in order to support the network that uses untraceable payment methods, such as Bitcoin.

As they monitor the log files, they observe that a Greek IP is distributing a video file which has been downloaded beforehand; this is called *seeding* in P2P protocols. After flagging this IP, authorities are noticing a constant connection for a long period of time and therefore it is very possible that the specific user is a regular supporter of this network and not just a random person who just happened to accidentally download an illegal file.

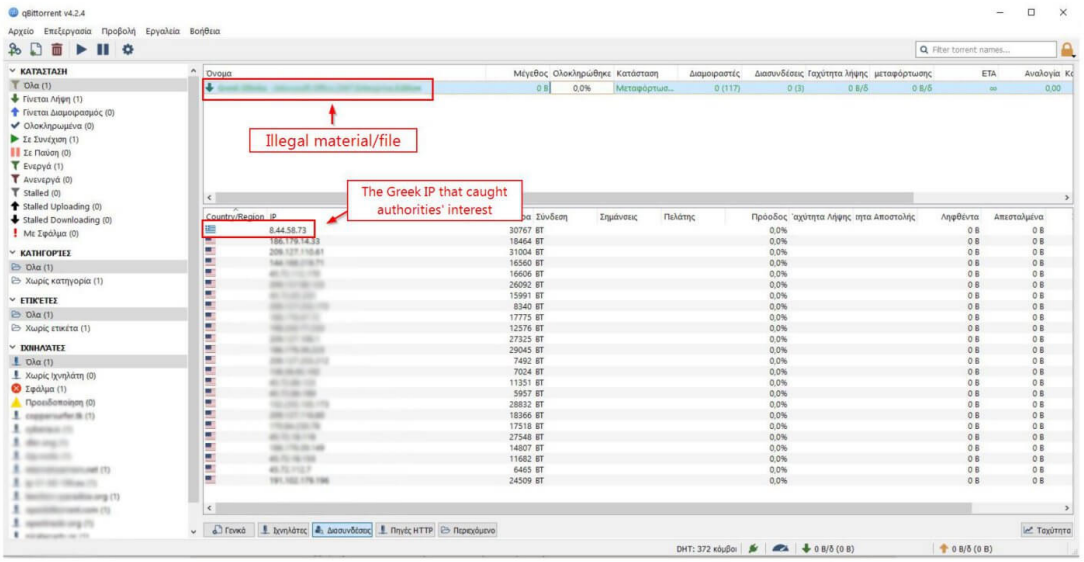


Figure 5.1: Screenshot of the P2P software used to distribute illegal pornographic material. The suspicious Greek IP which caught the authorities' interest is highlighted.

Meanwhile, authorities download the video in order to identify the person who is shown in the video and to validate if there is any formal complaint from this victim. Indeed, the minor person which is shown in the video has reported the leakage of the said material and since the victim is under-aged, authorities have now the right to catch the sexually deviant user and record the incident.

After contacting with the user's ISP, authorities manage to identify the person who is using the corresponding IP. They intrude into his house and after analyzing his hard

disks, they conclude that the suspect has downloaded over 10GB of pornographic material which involves kids and therefore he is suspended.

The responsible department inputs the incident into their systems in the form of free text: *A case of distributing illegal pornographic material which includes children committed by a sexually deviant user, named X.Y., conducted through Remote Access against the emotional integrity of an under-aged individual, resulted in emotional and moral harm of the victim.*

This description will be used by the Cybercrime Classification System in order to classify the incident to the corresponding Cybercrime type.

5.2 The Cybercrime Classification System in action

The Cybercrime Classification System steps in after authorities have recorded the Cybercrime Incident in free text. By using text mining and data mining techniques, it classifies the CI and provides useful information to the relevant stakeholders.

Since the CCS is already trained (Phase 1), the System executes Phase 2 (Classification-Prediction phase). In order to do so, the CCS must use text mining to extract the features of the CI, map the features to numbers (see tables 4.2 to 4.7) and then use this feature vector to finally classify the CI.

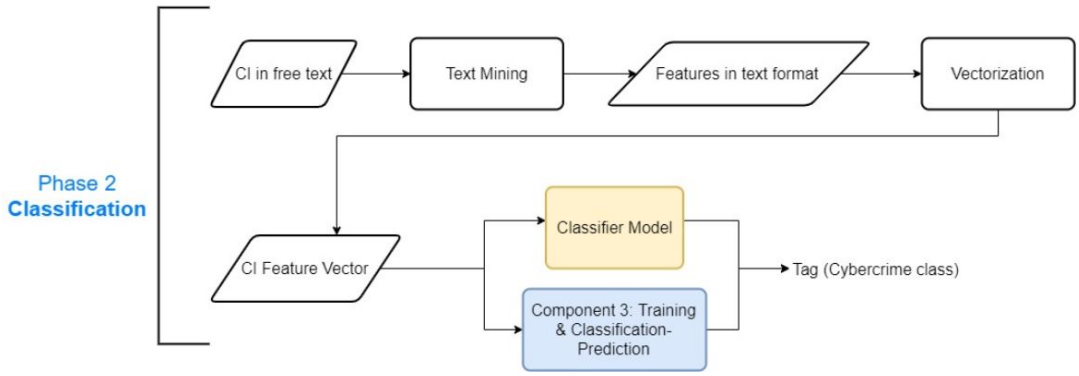


Figure 5.2: CCS Phase 2: Classification-Prediction.

Text mining — The proposed Cybercrime Classification System receives as input the CI in the form of free text: *A case of distributing illegal pornographic material which includes children committed by a sexually deviant user, named X.Y., conducted through Remote Access against the emotional integrity of an under-aged individual,*

resulting in emotional and moral harm of the victim.

Based on this description, the CCS applies text mining techniques in order to extract the features founded in the description. This process leads to the vector:

CI=[Sexually deviant user, Remote access, Emotional, Individual, Individual]

This vector can be represented also as a table:

Offender	**Access violation**	**Target**	**Victim**	**Harm**
Sexually deviant user	Remote access	Emotional	Individual	Individual

Vectorization (transformation of the CI to numerical format) — The CCS now receives the feature vector in text format and by using tables 4.2 to 4.7, it maps the features to numerical values. This process produces the following vector:

$$CI_x = [OF_{10}, AV_3, T_5, V_1, H_2]$$

This feature vector will be given to Component 3 in order to reach the final stage of Classification.

Classification-Prediction — Component 3 of CCS is responsible for receiving a feature vector and then classifying it to the correct class using the Naive Bayes classifier model, which has already been built during the Training phase.

Let's see how the CCS classifies the CI, with the help of Weka software:

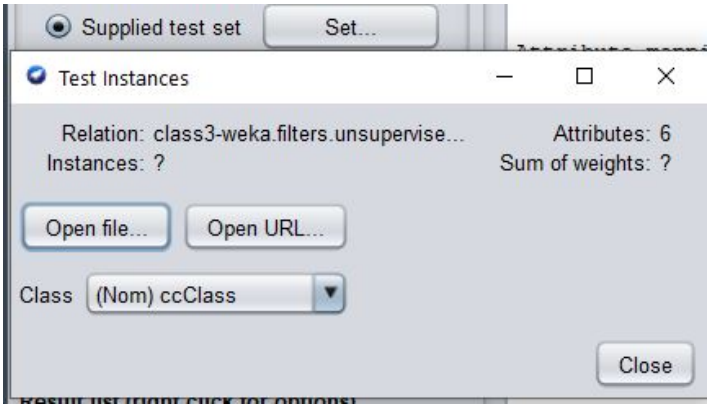
1. Create the CI instance to be given as input to Weka:

```
@relation class3-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last

@attribute offender {1,2,3,4,5,6,7,8,9,10,11,12}
@attribute accessviolation {1,2,3}
@attribute target {1,2,3,4,5}
@attribute victim {1,2,3}
@attribute harm {1,2,3,4}
@attribute ccClass {1,2,3,4,5}

@data
10,3,5,1,2,?
```

Note that *10,3,5,1,2,?* represents the relevant CI's features and *"?"* represents the unknown Cybercrime Class.

2. Select the Naive Bayes classifier:

3. Select the CI instance to be tested/classified:



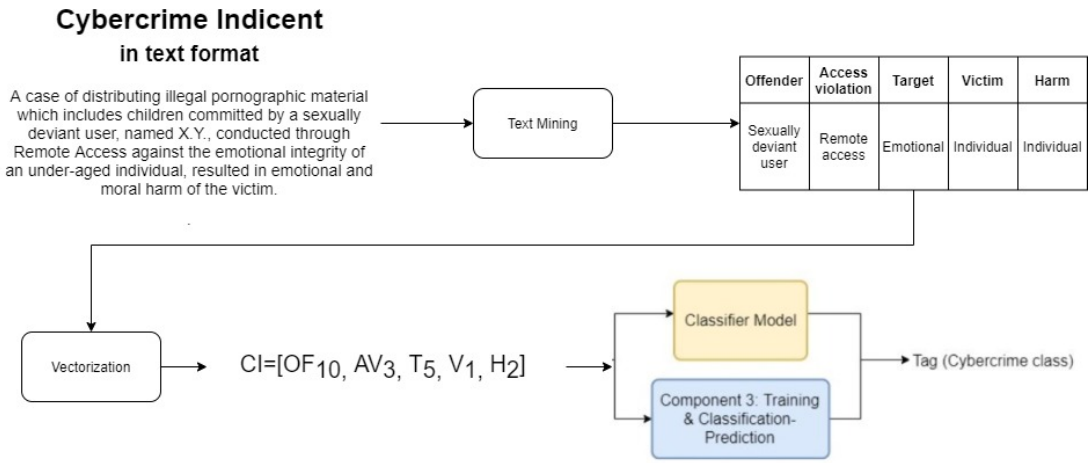
4. Run the Classification.

5. Results interpretation:

```
Attribute mappings:

Model attributes                    Incoming attributes
----------------------------        ----------------
(nominal) offender             --> 1 (nominal) offender
(nominal) accessviolation      --> 2 (nominal) accessviolation
(nominal) target               --> 3 (nominal) target
(nominal) victim               --> 4 (nominal) victim
(nominal) harm                 --> 5 (nominal) harm
(nominal) ccClass              --> 6 (nominal) ccClass


Time taken to build model: 0 seconds

=== Predictions on test set ===

    inst#     actual  predicted error prediction
        1        1:?        3:3       0.914
```

We can see that the CCS classified the instance to **Cybercrime Class 3**. This Classification is **correct**, since Cybercrimes such as downloading or sharing child pornography material belong to **Class 3: Content-related offences**, as shown in table 4.7. By looking at figure 3.3, which presents a two-layer Cybercrime Classification, we can see that Type C (equivalent to Class 3) includes *C2. Child Pornography*.

Concluding, the CCS managed to correctly classify the Cybercrime Instance and therefore produce useful information that will be thoroughly in the next section. Below is the CCS's flow for classifying the said CI:



5.3 Conclusions

Concluding, in this Chapter we present a (supposed) real-world Cybercrime Incident regarding possessing and distributing child pornography material. The Case Study focuses on all the aspects of the CI: from authorities catching the criminal to the execution of the CCS.

The CCS uses the Component 3 in order to classify the CI and quickly identify the type of Cybercrime that occurred. As we have already discussed, this helps to propose the appropriate counter-measures, to develop a more effective response policy and, in general, to confront the CI quicker and more systematically. Also, analysts can also use this data in order to find recurring patterns and correlations between the CI's features.

Chapter 6

Discussion & conclusions

The development of the ideas and of the proposed system have produced some interesting results. The main purpose of this thesis is to present a new concept of applying Data Mining techniques to the Cybercrime domain and contribute to the mitigation of the said phenomenon via Classification methods.

This chapter concludes this thesis and summarizes the outcomes and the research contribution of the proposed Cybercrime Classification System. In addition, several suggestions are made regarding future research directions.

6.1 Conclusions

The proposed Cybercrime Classification System aims towards the automated and systematic Classification of Cybercrime Incidents using Data Mining techniques. Based on the presented experiments using synthetically generated data, it can be concluded that such a system can achieve decent results and high accuracy regarding correct Classification. One of the main contributions of this thesis is to provide a novel method for the automated Classification of Cybercrime Incidents which could be used by the authorities in order to develop a holistic framework for handling such incidents.

As Cybercrime is growing exponentially, it is critical to come up with new and effective handling methods. The Cybercrime Classification System manages to assist the confrontation of such incidents as it can significantly reduce the time required to respond to Cybercrime Incidents and also produce useful insights that would help the authorities to prevent similar incidents from happening in the future.

In Chapter 4, where the Cybercrime Classification System is presented, it is made clear that there is a lack of publicly available data regarding Cybercrime Incidents, but with the proper use of the system's components, an equivalent dataset can be generated using techniques such as Synthetic Data Generation. This approach helps the analysts to adjust the data according to their specifications and requirements, and therefore accomplish to simulate real-world incidents.

The results of the proposed system can contribute towards the effective prevention, confrontation and prediction of Cybercrime Incidents by the authorities. As the previous chapters made obvious, the automated process of Cybercrime Incident Classification can enhance the whole process of handling these incidents, since it can produce statistical data regarding Cybercrime, help analysts to detect possible correlations between incidents, propose appropriate response measures according to an existing response policy and also assess the threat severity of each Cybercrime.

6.2 Future work

Although we consider that the purposes of this thesis have been accomplished, several improvements and extensions could be done in order to achieve better results and provide more functionalities.

1. *Layer-2 Classification*: The proposed system classifies Cybercrime Incidents based on Layer 1, as shown in figure 3.3. A significant improvement that could be made is to classify Cybercrime occurrences based on Layer 2 in order to identify the class more specifically. Layer 1 is a broader layer and therefore refers to more general categories, while Layer 2 expands the types more explicitly.

   This thesis initially aimed at achieving a Layer 2 Classification but many obstacles came up and we did not manage to define the slight differences between Layer's 2 classes. However, it is an interesting challenge that future researchers may want to overcome and further improve the accuracy of the Cybercrime Classification System.

2. *Improvements regarding the Synthetic Dataset*: As already stated, real-world data regarding Cybercrime Incidents do not publicly exist. Therefore, it is needed to create our own data in order to apply the required Data Mining techniques.

For the generation of the presented synthetic dataset, in summary, we followed this approach: 1) define the attribute values applicable to each class, 2) create a programming script which generates a *csv* file that includes all the Cybercrime Incidents combinations for each class, based on the defined applicable values.

The produced dataset contains valuable information that is required for the Training phase of our Classification System. Future work could include a thorough analysis of correlations between the Cybercrime features and Cybercrime types and define an unambiguous framework that decides if a certain feature value can be related to the respective Cybercrime class. Since bibliography does not provide such information, researchers could possibly analyze real-world incidents from various sources and define the correlations based on unbiased assumptions, unlike the assumptions made on this thesis.

3. *Develop a GUI based on the proposed system*: The proposed system, in order to achieve the Classification of new incidents, uses the Weka software which requires data mainly in *csv* format. So, if we want to classify a new instance we need to create a *csv* file which contains the instance in a vector-like format and then give it as input to Weka.

   To simplify this process, future work could include developing a Graphical User Interface as a standalone program. Although the underlying code would still be based on the principles of the proposed Cybercrime Classification System, a GUI with text fields and other graphical components could speed up the process of classifying Cybercrime Incidents and also provide ease of use for novices.

4. *Text Mining process*: This thesis presents the concept of Text Mining during Classification that transforms the free-text description of the occurred Cybercrime Incident (as recorded by the authorities) into a feature vector. This process is mandatory for Classification since the classifier model needs feature vectors in order to work properly and achieve class prediction.

   The Text Mining process is not analyzed in this thesis because it requires extensive research which is outside of the thesis' scope. Therefore, researchers could extend the practical functionality of the proposed system by applying actual Text Mining techniques that can transform the free-text description of an incident into

a feature vector which represents the same incident.

5. *Further improve the Classification accuracy by developing a customized Classification algorithm*: In this thesis a comparison of the most common used Classification algorithms is conducted and finally the *Naive Bayes classifier* is selected for the Cybercrime Classification System. Usually, in Classification problems, researchers may want to modify an existing algorithm in order to improve its efficiency and/or accuracy and adjust it to the nature of the problem. It is obvious that maximizing efficiency and accuracy is always desirable in a Classification problem.

6. *Predict Cybercrime based on statistical data*: By constantly training the Classification model based on new data and therefore acquiring a big enough dataset, analysts could perform several statistical analyses in order to gain knowledge and insight. For instance, the Authorities could focus on an offender who conducted a specific Cybercrime type and then find a similar offender. If the latter offender has committed another crime, analysts could assume that the former offender may conduct a similar crime in the future.

7. *Apply other Data Mining techniques*: A significant contribution of this thesis is to highlight how Data Mining could help the mitigation of Cybercrime. Although we mainly focus on Classification, the presented architecture could also be used for applying techniques such as:

    - *Clustering*: Clustering could help to group together similar offences and analyze their features similarities and come to several conclusions.
    - *Association Rules*: ARs could assist the Authorities to better identify the correlations between Cybercrime's features. For example, ARs could produce an Association Rule which would prove that when a CI has a specific feature value, it is very possible to has an other specific feature value (*Feature 1* $\longrightarrow$ *Feature 2* translates, in short, to "when Feature 1 happens, also Feature 2 happens").

6.3 Summary

This thesis extends the Cybercrime Incident Architecture by proposing a feature-based Cybercrime Classification System which uses Data Mining techniques in order to classify new Cybercrime Incidents. Since the Classification of Cybercrimes, as explained, is useful to the authorities, it is obvious that such a system would assist the mitigation of the phenomenon by properly analyzing the produced insight. A Case Study is also presented and it tracks the whole process: from the authorities recording the incident to the final Classification by the CCS. The results are satisfactory and indicate that a practical use case of the proposed system is feasible and important. Future work could further improve the functionalities of the CCS in order to provide more utilities.

References

[1] George Tsakalidis et al. “A cybercrime incident architecture with adaptive response policy”. In: *Computers & Security* 83 (2019), pp. 22–37.

[2] Eoghan Casey. *Digital evidence and computer crime: Forensic science, computers, and the internet*. Academic press, 2011.

[3] Council of Europe. *The commission communication ”towards a general policy on the fight against cyber crime”*. 2007.

[4] US Department of Justice. *Computer Crime and Intellectual Property Section*. 1996.

[5] Graham Roderick. *The Difference between Cybersecurity and Cybercrime and Why It Matters*. 2017.

[6] J.B. Hill and N.E. Marion. *Introduction to Cybercrime: Computer Crimes, Laws, and Policing in the 21st Century*. Praeger Security International Series. Praeger, 2016. ISBN: 9781440835339.

[7] Savita Mohurle and Manisha Patil. “A brief study of wannacry threat: Ransomware attack 2017”. In: *International Journal of Advanced Research in Computer Science* 8.5 (2017), pp. 1938–1940.

[8] David Buil-Gil et al. “Cybercrime and shifts in opportunities during COVID-19: a preliminary analysis in the UK”. In: *European Societies* 23.sup1 (2021), S47–S59.

[9] Statista. *Cyber Crime & Security | Statista*. www.statista.com/markets/424/topic/1065/cyber-crime-security. (Accessed on 06/04/2021).

[10] Morvareed Bidgoli and Jens Grossklags. “End user cybercrime reporting: what we know and what we can do to improve it”. In: *2016 IEEE International Conference on Cybercrime and Computer Forensic (ICCCF)*. IEEE. 2016, pp. 1–6.

[11] Lawrence E Cohen and Marcus Felson. “Social change and crime rate trends: A routine activity approach”. In: *American sociological review* (1979), pp. 588–608.

[12] Jonathan Clough. *Principles of cybercrime*. Cambridge University Press, 2015.

[13] J Johnson. *Worldwide digital population as of January 2021*. 2021.

[14] Mohamed Chawki. “Anonymity in cyberspace: Finding the balance between privacy and security”. In: *International Journal of Technology Transfer and Commercialisation* 9.3 (2010), pp. 183–199.

[15] Georgios Tsakalidis. “Camco a framework for classification, analysis & monitoring cybercrime-related offences”. In: (2016).

[16] Sinchul Back and Jennifer LaPrade. “The future of cybercrime prevention strategies: human factors and a holistic approach to cyber Intelligence”. In: *International Journal of Cybersecurity Intelligence & Cybercrime* 2.2 (2019), pp. 1–4.

[17] Seong-Sik Lee et al. “A test of structural model for fear of crime in social networking sites”. In: *International Journal of Cybersecurity Intelligence & Cybercrime* 2.2 (2019), pp. 5–22.

[18] Elmarie Kritzinger and Sebastiaan H von Solms. “Cyber security for home users: A new way of protection through awareness enforcement”. In: *Computers & Security* 29.8 (2010), pp. 840–847.

[19] Dilip Chenoy and Rahul Rishi. *Confronting the new-age cybercriminal*. 2019.

[20] Charu C Aggarwal. *Data mining: the textbook*. Springer, 2015.

[21] Abhinav Rai. *Association Rule Mining: An Overview and its Applications*. www.upgrad.com/blog/association-rule-mining-an-overview-and-its-applications/. (Accessed on 06/10/2021). 2019.

[22] Ben Lutkevich. *What are Association Rules in Data Mining (Association Rule Mining)?* https://searchbusinessanalytics.techtarget.com/definition/association-rules-in-data-mining. (Accessed on 06/10/2021). 2020.

[23] Victor Roman. *Unsupervised Machine Learning: Clustering Analysis*. www.towardsdatascience.com. (Accessed on 06/10/2021). 2019.

[24] Sophia G Petridou et al. "Time-aware web users' clustering". In: *IEEE Transactions on Knowledge and Data Engineering* 20.5 (2008), pp. 653–667.

[25] Sophia G Petridou et al. "On the use of clustering algorithms for message scheduling in WDM star networks". In: *Journal of Lightwave Technology* 26.17 (2008), pp. 2999–3010.

[26] Christos K Liaskos et al. "Clustering-driven wireless data broadcasting". In: *IEEE Wireless Communications* 16.6 (2009), pp. 80–87.

[27] Vassiliki Koutsonikola et al. "Correlating time-related data sources with co-clustering". In: *International Conference on Web Information Systems Engineering*. Springer. 2008, pp. 264–279.

[28] Zhexue Huang. "Extensions to the k-means algorithm for clustering large data sets with categorical values". In: *Data mining and knowledge discovery* 2.3 (1998), pp. 283–304.

[29] *Data Clustering Algorithms - k-means clustering algorithm*. www.sites.google.com/site/dataclusteringalgorithms/k-means-clustering-algorithm. (Accessed on 06/10/2021).

[30] *K Means Clustering Simplified in Python | K Means Algorithm*. www.analyticsvidhya.com/blog/2021/04/k-means-clustering-simplified-in-python/. (Accessed on 06/10/2021). 2021.

[31] Mohammed J. Zaki and Wagner Meira. *Data mining and machine learning: fundamental concepts and algorithms*. Cambridge, United Kingdom ; New York, NY: Cambridge University Press, 2020. ISBN: 978-1-108-47398-9.

[32] Laszlo Kozma. "k Nearest Neighbors algorithm (kNN)". In: *Helsinki University of Technology* (2008).

[33] Chao Li et al. “Using the K-nearest neighbor algorithm for the classification of lymph node metastasis in gastric cancer”. In: *Computational and mathematical methods in medicine* 2012 (2012).

[34] *K-Nearest Neighbor(KNN) Algorithm for Machine Learning*. www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning.

[35] Ravi Kothari and Ming Dong. “Decision trees for classification: A review and some new results”. In: *Pattern recognition: from classical to modern approaches* (2001), pp. 169–184.

[36] Badr Hssina et al. “A comparative study of decision tree ID3 and C4. 5”. In: *International Journal of Advanced Computer Science and Applications* 4.2 (2014), pp. 13–19.

[37] Esperanza García-Gonzalo et al. “Hard-rock stability analysis for span design in entry-type excavations with learning classifiers”. In: *Materials* 9.7 (2016), p. 531.

[38] K SRIVASTAVA Durgesh and B Lekha. “Data classification using support vector machine”. In: *Journal of theoretical and applied information technology* 12.1 (2010), pp. 1–7.

[39] Siddharth Misra, Hao Li, and Jiabo He. *Machine learning for subsurface characterization*. Gulf Professional Publishing, 2019.

[40] Ana C Lorena et al. “Comparing machine learning classifiers in potential distribution modelling”. In: *Expert Systems with Applications* 38.5 (2011), pp. 5268–5275.

[41] Amanpreet Singh, Narina Thakur, and Aakanksha Sharma. “A review of supervised machine learning algorithms”. In: *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*. Ieee. 2016, pp. 1310–1315.

[42] Nahla Ben Amor, Salem Benferhat, and Zied Elouedi. “Naive bayes vs decision trees in intrusion detection systems”. In: *Proceedings of the 2004 ACM symposium on Applied computing*. 2004, pp. 420–424.

[43] Pumitara Ruangthong, Pradit Songsangyos, and Soontaree Kankaew. “Solving Imbalanced Problem of Muticlass Data Set with Class Balancer and Synthetic Minority Over-sampling Technique”. In: *International Journal of Applied Computer Technology and Information Systems* 6.1 (2016), pp. 87–90.

[44] Salvador García, Julián Luengo, and Francisco Herrera. *Data preprocessing in data mining*. Vol. 72. Springer, 2015.

[45] Richard J Roiger and MW Geatz. “Data Mining: A Tutorial-based Primer, Pearson Education”. In: *Inc: USA* (2003).

[46] *ClassBalancer (Weka API Documentation)*. https://weka.sourceforge.io/doc.dev/weka/filters/supervised/instance/ClassBalancer.html. (Accessed on 05/21/2021).

[47] Victoria López et al. “An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics”. In: *Information sciences* 250 (2013), pp. 113–141.

[48] Ian Witten. *www.cs.waikato.ac.nz/ml/weka/mooc/dataminingwithweka*. www.cs.waikato.ac.nz/ml/weka/mooc/dataminingwithweka. (Accessed on 05/31/2021).

[49] George Tsakalidis and Kostas Vergidis. “A systematic approach toward description and classification of cybercrime incidents”. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 49.4 (2017), pp. 710–729.