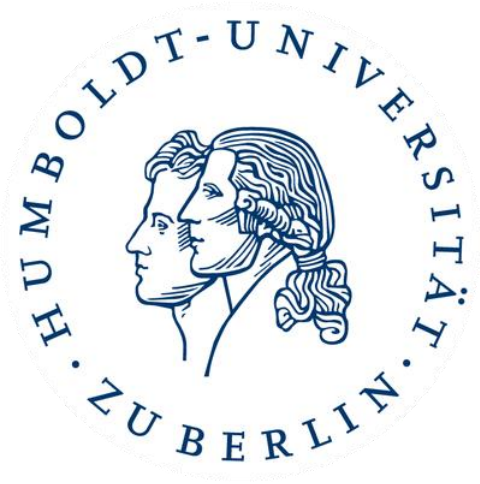
Chair of Information Systems at Humboldt-Universität zu Berlin



Cyber-Attack Identification using Machine Learning

Seminar Paper

in IT Security and Privacy

at the School of Business and Economics
of Humboldt-Universität zu Berlin

Submitted by:

Surname: Velev
First name: Georg
Matriculation number: 601868
E-Mail: velevgeo@hu-berlin.de

Surname: Pekova
First name: Iliyana
Matriculation number: 598458
E-Mail: pekovail@hu-berlin.de

Abstract

The problematic of cybercrimes becomes more and more relevant as increasing amounts of vulnerable information are stored in the cyberspace and financial assets are operated virtually. Therefore, this work focuses on the identification of cyber-attacks using data mining techniques. The data is obtained from the official website of Knowledge Discovery and Data Mining. The methodology of the Cross-Industry Standard Process for Data Mining (CRISP-DM) is followed throughout the analysis and implemented in the open source software Python. Two machine learning algorithms, Random Forest and Bernoulli Naïve Bayes, are applied, and their performance is evaluated in terms of particular metrics. The analytical model that achieves better results is selected for the deployment phase.

List of Contents

List of Tables v
List of Figures vi
List of Abbreviations vii
1 Introduction 1
2 Theoretical background 1
2.1 Regular Network Activity versus Cyber-Attack 1
2.2 Network Protocols and Services 3
2.2.1 Hypertext Transfer Protocol 3
2.2.2 Virtual Private Networks 3
2.2.3 Simple Mail Transfer Protocol 4
2.2.4 Domain Name System 4
2.2.5 Transmission Control Protocol 4
2.2.6 User Datagram Protocol 4
2.2.7 Internet Control Message Protocol 5
2.2.8 Echo and Echo Reply 5
2.2.9 Protocol Unreachable 5
3 Related Research 5
4 Methodology: CRISP-DM 6
4.1 Business Understanding 6
4.2 Data Understanding 7
4.3 Data Preparation 7
4.4 Modelling 8
4.4.1 RF 8
4.4.2 BNB 9
4.5 Evaluation 10
4.6 Deployment 11
5 Empirical Research 11
5.1 Data Retrieval 11
5.2 Data Analysis 12
5.2.1 Duplicated Observations 12
5.2.2 Missing Values 13
5.2.3 Class Distribution of the Target Variable 13
5.2.4 Encoding categorical Features 14
5.2.5 Feature Selection 16

5.3 Modeling and Evaluation 17
5.4 Discussion and Analysis of the Results 18
6 Conlusion 18
7 Reference List 20
8 Appendix 25
8.1 Table Appendix 25
8.2 Figure Appendix 29

List of Tables

Table 1: Related Research 27
Table 2: Confusion Matrix 27
Table 3: Evaluation Metrics 28
Table 4: Types of Cyber-Attacks 28
Table 5: Summary Correlation Filter BNB 28
Table 6: Tested Hyperparameters RF 28
Table 7: Tested Hyperparameters BNB 29

List of Figures

Figure 1: Missing Values 29
Figure 2: Values Count Target Variable 29
Figure 3: Categories of the transformed Target Variable 30
Figure 4: Average Duration Cyber-Attacks 30
Figure 5: Countplot Flag 31
Figure 6: Proportions Protocol_Type 31
Figure 7: Stripplot serror_rate, protocol_type and Target 32
Figure 8: 3D Count plot TCP, service and Target 32
Figure 9: 3D Count plot UDP, service and Target 33
Figure 10: 3D Count plot ICMP, service and Target 33
Figure 11: Correlation Filter RF 34
Figure 12: Correlation Filter BNB without Binarization 34
Figure 13: Correlation Filter BNB with Binarization 34
Figure 14: Parameter Optimization RF 35
Figure 15: Parameter Optimization BNB 35

List of Abbreviations

ACK	Acknowledgement
AUC	Area under the ROC Curve
BNB	Bernoulli Naive Bayes
CART	Classification and Regression Tree
CRISP-DM	Cross Industry Standard Process for Data Mining
DNS	Domain Name System
DoS	Denial of Service
DR	Detection Rate
FN	False Negative
FNR	False Negative Rate
FP	False Positive
FPR	False Positive Rate
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IDS	Intrustion Detection System
IP	Internet Protocol
KDD	Knowledge Discovery in Databases
LAN	Local Area Network
OOB	Out-of-Bag
OSI Model	Open System Interconnection Model
R2L	Root to Local
REJ	Rejected
RF	Random Forest
ROC	Receiver operating Characteristics
SF	Normal Status
SMOTE	Synthetic Minority Oversampling TEchnique
SMTP	Simple Mail Transfer Protocol
SYN	Synchronization
TCP	Transmission Control Protocol
TCP/IP Model	Transmission Control Protocol/Internet Protocol Model
TN	True Negative

TP	True Positive
TPR	True Positive Rate
U2R	User to Root
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VPN	Virtual Private Network

1 Introduction

A brief look in the cybercrime history for the past 4 years provides examples of multiple massive attacks which cost companies millions of dollars as well as their reputation among users due to billions unities of stolen personal data. Some very famous ones are WannaCry (2107), NotPetya (2016), Equifax (2017), the massive denial-of-service attack on GitHub (2018). The statistics regarding cybercrimes also prove that the issue is reaching new heights and thus continuously gaining importance. The count of cyber-attacks in the 2nd quarter of 2018 shows an increase of 47% compared to the previous year. On the cost side, the expenses caused by cyber incidents in 2018 amount to the total sum of $1.5 trillion. Hence, an urgent necessity to recognize potential cyber threats has arisen, in order to prevent the actual realization of intrusions (Fruhlinger 2018).

Data mining techniques that benefit from the overwhelming amount of information stored in the network activity records of the companies can be used to discover hidden patterns in the data (Prasad and Madhavi 2012). In this context, machine learning represents a field in artificial intelligence that applies particular algorithms which learn from the observations in the past and generate predictions about the future (Langley and Simon 1995). Therefore, this kind of models identify relationships between the variables in a particular dataset and use the knowledge gained during the training process to provide a solution for the problem that is being examined.

This paper aims at building an analytical model for the identification of cyber-attacks by using machine learning algorithms. The work consists of six parts. Firstly, the relevant theoretical background is provided. Then, a literature review summarizes the results of related studies. Afterwards, Section 4 presents the six phases of the Cross Industry Standard Process for Data Mining (CRISP-DM) model. Section 5 contains the empirical research. Finally, Section 6 consists of a summary as well as several recommendations for the model deployment.

2 Theoretical background

2.1 Regular Network Activity versus Cyber-Attack

From the physical point of view, regular network activity consists in transferring data stored in data packets from the client (sending host) to the host server (receiving host) through routers and switchers. From a logical perspective the data transfer is enabled and continuously supported by multiple network protocols, which are in charge of different tasks. Each network protocol belongs to a certain networking layer. The latter can be classified according to two alternative models: the TCP/IP model and the OSI model. The OSI model is a standard

reference model, which defines seven types of networking layers. It provides a standardized scheme of how the different components, enabling network communication, should interact and divide functionalities. In addition, the model assumes that any functionality an application could need is already available within the model. By contrast, the TCP/IP model was designed to solve specific problems and drops the assumption that all functionalities are predefined by the model. It distinguishes between four layers, which are an application, a transport, an internet and a network one (Burke 2017).

The current work adheres to the TCP/IP model. All of the networking layers contribute to the data encapsulation, which means adding a new header to the data packets as they are being transferred through the various layers. By doing so, each data packet can be distinguished from the others and clearly identified by the layer it is currently passing through. One of the most significant protocols - the Internet Protocol - has the function of transferring the data packets to the right destination. This happens by assigning numerical combinations (an IP address) to each device in the network. This way, the data can "travel by" the devices, which do not have a particular IP address until it has reached the correct one.

This research examines four particular types of cyber-crimes: Denial of Service (DoS), Remote to Local (R2L), User to Root (U2R) and Probing. DoS attacks affect the functionality of a certain service by making it of limited access for the users or blocking it completely. Therefore, these cyber-threats do not aim at stealing any personal information. They are often used as a distraction measure in order to perform more harmful attacks. Consequently, DoS attacks cause monetary and time costs for the firms. A typical example is the malfunctioning of e-mail services. In comparison to that, R2L attacks aim at gaining access to the machine of the victim. Hence packets are being sent over the network in order to exploit some vulnerabilities of that particular system. A common example is password guessing. In U2R intrusions the attacker first sets up a legitimate network connection on the remote side with a normal user account and then attempts to obtain the privileges of a root user or an administrator by illegally trying to take advantage of a local system's vulnerability. The fourth attack type – Probing – refers to a cyber-crime which aims at scanning a particular network in order to collect and/or monitor information about the network activity and identify vulnerabilities that can be abused later on (Ashraf et al., 2018; Siddiqui and Naahid, 2013; Upadhyay and Singh, 2015).

To provide a clear comparison between a regular network activity and a cyber-attack, an example of a typical phishing attack will be presented briefly. Some of the most frequent phishing intrusions are the ones based on the Man-in-the-Middle principle. The main idea of this principle is to create a proxy server and afterwards to redirect the user to it instead of letting

the data reach the real one, without any signs or indications of doing so. In turn, the data packet sent by the user does not arrive at its initial destination but lands at the fake server, created by the hacker, enabling the latter to monitor and record the activity of the victim and collect any vulnerable data. This kind of attack is usually connected to an intrusion into the transport layer, since it is the one establishing a private, encrypted connection between the user and the receiving host.

2.2 Network Protocols and Services

This section provides brief information regarding the most relevant protocols and services in the context of the current work.

2.2.1 Hypertext Transfer Protocol

The HTTP belongs to the application network layer. It serves mostly, but not solely, for the transfer of webpages from a webserver (host) to a web browser (client) within an IP network, which is known as the Client-Server-Principle. An HTTP-request, sent by the browser, initializes the interaction and the HTTP-response from the server terminates it. The communication is text-based. What practically enables the client to address particular content on the server by sending a request is the URL (Uniform Source Locator). It consists of the http:// abbreviation, the domain name, the exact path of the desired content and the name of the directory, storing the content. The server in turn responds by providing the demanded data. In case of an unspecified or a non-existing directory, the server provides the default content or generates an error respectively. Finally, it is important pointing out that the basic HTTP-connection is not encrypted. Thus, virtual private networks (VPNs) are established as a solution to potential security-related problems (Luber, 2018b; Rouse, 2014).

2.2.2 Virtual Private Networks

Virtual private networks can be seen as an encrypted modification of the original HTTP. VPNs enable the creation of externally non-transparent connection tunnels. These serve for protecting the transferred content from the risks of the cyber space, since the public Internet – instead of cost intensive physical networks – is the medium of communication and data-exchange. A VPN-connection can be point-to-point, point-multipoint or without a strictly predefined structure. A typical example for a VPN is a closed company or department network, which is only available for authorized members and is accessible anytime regardless of a user's location (home office e.g.) (Luber 2018a).

2.2.3 Simple Mail Transfer Protocol

The SMTP belongs to the application network layer. As the name suggests, the protocol's functionality is the e-mail transfer within an IP network. Similarly to the HTTP, the SMTP is a client-server based protocol. Alternatively, the connection can be between two SMTP-servers. In this case, one of them is considered the client. The client is in charge of establishing connection to the server and afterwards sending it instructions and information regarding the transaction that must take place. The server in turn executes the transaction and responds to the client by providing the result (Luber 2018c).

2.2.4 Domain Name System

The DNS is a protocol operating within the application network layer. DNS uses UDP as a transfer protocol by default and only switches to TCP if the volume of the transferred data exceeds 512 or 4096 bytes without and with a DNS-extension respectively. Technically, DNS is a service transforming the numerical IP addresses of domains into natural language. This way, the end user does not need to know and enter the corresponding IP address of a website to access it, but only its name in the form of an URL. In addition, a DNS allows a change of the IP-address of a website, without it affecting the end user in any way, since the latter only uses a domain's name in the form of natural language (Srocke, 2018; Strotmann, 2019).

2.2.5 Transmission Control Protocol

The TCP is a transport layer protocol. The communication in the case of the TCP is connection-oriented, which implies establishing an arranged communication session over a fixed data channel. The TCP is suitable, when high reliability is required. The danger of lost or unsent data is practically inexistent because of the TCP's error recovery property, meaning retransmission if necessary. Furthermore, by being a connection-oriented protocol, the TCP guarantees that the transferred data will arrive and it will be arranged and if needed rearranged in the order specified by the sending host. On the negative side, a TCP is always relatively slow in its performance due to all control and quality check it provides (Ahmed Elnaggar 2015).

2.2.6 User Datagram Protocol

Like the TCP, the UDP belongs to the transport network layer. However, the UDP implies connectionless communication, which involves neither a prior arrangement nor a fixed data channel. Instead, each data packet is addressed individually. The use of the UDP is mostly reasonable, when no high reliability is expected, since the protocol does not have any control or recovery functions. Consequently, the transmitted data may be lost, unsent or sent in the

wrong order. Similarly to the TCP, there is a trade-off between speed and reliability. The UDP is much faster, since it does not ensure the completeness of the data (Ahmed Elnaggar 2015).

2.2.7 Internet Control Message Protocol

The ICMP is an internet layer protocol. Usually it is not used for data exchange, but has a rather supportive function, namely sending error messages, when a server is not available or a host cannot be reached (Arkin 2000).

2.2.8 Echo and Echo Reply

The echo and the echo reply are both ICMP services. They are also known as ping messages, which are signals sent between two hosts to check the connectivity stand between two devices. It starts with one of the hosts sending a ping message to the other one. A ping message consists of a single packet, containing an echo request and serves the following purposes: checking the opposite party's availability and measuring the response time. The echo request is in turn followed by an echo reply, which is a single packet, as well. For security reasons, some companies have their echo requests disabled, which makes it harder for hackers to discover and respectively access the system (Arkin 2000).

2.2.9 Protocol Unreachable

Another ICMP destination is the protocol unreachable message, which is the source host receives when the destination host cannot be reached. From a technical point of view, this is the case, when the appointed transport protocol is not supported (Arkin 2000).

3 Related Research

When implementing a machine learning algorithm one can mainly distinguish between a classification problem and an anomaly detection problem. In this context, it is essential to differentiate between supervised, semi-supervised and unsupervised learning. The first implication of supervised learning is the presence of a target variable, which is the variable to be explained by the model. Furthermore, supervised learning means that the target variable is labeled, meaning that each observation belongs to a particular predefined class. The applied algorithm in turn learns from the labeled train set and then assigns each of the observations in the test set to a certain class, which is the process of classification. The target variable in classification problems is always categorical, unlike the continuous one in a regression problem, which is another supervised learning alternative. By contrast, unsupervised learning has neither a dependent variable to be predicted nor labels. The purpose of the algorithm is to discover hidden patterns in the data and to identify anomalous observations – observations,

which have not been seen up to the current point. Thus, unsupervised learning refers to an anomaly detection problem. Finally, semi-supervised learning is also a form of anomaly detection, but performed on a partially labeled dataset (Choudhary, 2017; Naser, 2017).

It is of high importance choosing the right technique when handling a machine learning problem. There are certain aspects to consider in this context. If the relevant dataset shows a strong imbalance and hence contains a small number of anomalies and many negative (=normal) observations, anomaly detection techniques are used, since the algorithm does not have enough positive (=anomalous) input to learn from. Anomaly detection approaches are also opted for, when there are many different types of anomalous observations and the algorithm cannot learn what the anomalies look like based on the few positive examples. This is also partially justified by the fact that future anomalies may have no similarities to the ones which are already known. On the other hand, supervised learning is preferable, when the dataset contains a large number of negative and positive examples. This enables the algorithm to learn how an anomaly could possibly look like. In addition, in the case of numerous positive examples the probability for a newly appeared anomaly to be similar to some of the already identified ones, is higher (Artificial Intelligence - All in One 2017).

The focus of the current paper is placed on recognizing malicious network activity and identifying cyber-attacks. Under certain circumstances, the machine learning problem handling such topic could be defined as an anomaly detection one. This is the case when the analyzed data contains few observations of cyber-attacks and no clear target variable. However, if there are many positive examples and respectively multiple classes to which the observations of the target variable could be assigned, it is reasonable to perform classification. Based on the conditions referring to each alternative, the problem in the current work will be approached by the use of a classification machine learning algorithm.

4 Methodology: CRISP-DM

The CRISP-DM model provides a very structured and also agile approach in the context of business analytics which consists of six phases (Wirth and Hipp 2000).

4.1 Business Understanding

Business understanding is the first and therefore the most important phase in the entire data mining project. It focuses on the precise definition of the business objective which is used to define the data mining goals from a technical perspective (Shearer 2000). In the context of cyber attacks, institutions aim at increasing their protection from network security threats. This goal pre-determines the kind of data that should be collected and used for the training of the machine

learning algorithms so that they can identify cyber attacks in advance. Therefore, it is necessary to gain knowledge of the background of the institution that requires a particular solution. It is also recommended to prepare a project plan and to assess the situation regarding available resources, underlying assumptions, possible risks as well as arising expenses (Wirth and Hipp 2000).

4.2 Data Understanding

The second phase begins with the retrieval of relevant data. Furthermore, the main characteristics of the collected data have to be described. This includes, for example, the total number of observations, the type of the variables and their possible values. In addition to that, it is advisable to use visualization tools in order to explore the distribution of potentially important features. This may result in discovering some hidden patterns. In this context, it is also essential to verify the quality of the collected data by identifying the columns that contain missing values and duplicated observations. Moreover, the entire dataset has to be examined with respect to features providing similar, unnecessary or even unplausible information (Shearer 2000).

4.3 Data Preparation

The performed analysis is followed by data preparation. This implies handling missing values, data cleaning, deriving new variables from the already existing ones as well as performing feature selection (Wirth and Hipp 2000). In addition to that, methods that aim at reducing the imbalanced distribution of the class variable should also be applied in case the machine learning algorithms fail to identify the instances from the minority class (Kotsiantis et al. 2006).

To begin with, the underlying cause for the lacking information has to be established. Afterwards particular methods dealing with missing values have to be applied. In this context, numeric data that is lacking can be replaced with the mean or the median of the particular feature. By contrast, the most frequent occurrence can be used in the case of categorical columns. Furthermore, constants as well as dummy values are another option for the replacement. Moreover, missings can be imputed by the predicted values generated by a particular analytical model or by the group values resulting from k-means clustering. In addition to that, observations lacking too much information can be deleted. Entire columns that have a lot of missing samples can also be removed (Brownlee, 2017; Chopra, 2018; Rohrer, 2015).

Once the missing values are handled, it is essential to clean the data from outliers that are not representative of the examined problem (Brownlee 2018). These are objects that deviate

significantly from the rest of the values of a particular variable (Santoyo 2017). They usually have a negative impact on statistical analyses as they strongly influence measures like the variance as well as the mean of the features. Hence, they contribute to the asymmetric distribution of the data (Kwak and Kim 2017). Outliers resulting from an error made for example during the collection or the entry of the data should either be corrected in case the information necessary for that is available or removed. By contrast, outlying data points that are related to the variability of the samples of a particular feature require special attention as these may hold potentially interesting as well as important information (Osborne and Overbay 2004).

After the instances that are abnormally distant to other observations are dealt with in a way that is reasonable for the conducted analysis, it is recommended creating new features by using the already available ones. The underlying reason for that is that the derived attributes may turn out to be more valuable for the prediction of the target variable than the original ones (Ray 2016). Furthermore, feature selection has to be performed in order to choose the most optimal subset of variables. In this context, a correlation-based filter can be applied in order to remove the attributes that highly correlate with each other as this implies that they are holding redundant information (Hall 2000).

The last step of the phase devoted to data preparation focuses on handling class imbalance. Techniques like random undersampling and oversampling as well as SMOTE are frequently used methods that deal with this problem (Anil Kumar and Ravi, 2008; Batista et al., 2004).

4.4 Modelling

In the modeling phase, several models are applied and their hyperparameters are optimized (Wirth and Hipp 2000). This section focuses on the machine learning algorithms RF and BNB.

4.4.1 RF

RFs represent an ensemble of many CART classifiers combined into a single model in order to generate the prediction for a particular data mining problem (Breiman 2001). The CART algorithm recursively implements a binary split in order to partition the feature space by passing the data from the parent-nodes to the child ones until the end of the tree is reached (Lawrence and Wright 2001). Therefore, each root node is represented by each of the input attributes. The training process aims at generating homogenous terminal nodes where no further splitting is possible. In addition to that, RF grows a number of trees by bootstrap sampling the initial data (Chandradevan 2017). This means that a random sample of a particular size is taken with replacement from the provided data. In this way, bootstrap subsets of the variables are randomly

generated and used in order to build each tree. Therefore, the algorithm may select the same sample a couple of times and completely ignore some other ones (Belgiu and Drăguţ 2016). In addition to that, the bootstrap sampling can be regarded as the first stage of randomization in the learning process of RF (Nguyen et al. 2013). The second one is related to the splitting that takes place at each node of the generated trees. Thus, RF draws a random subset of features that are used in order to find the best possible split for the respective node. Once each of the trees in the forest is fully grown, then the information from the terminal nodes is aggregated by using majority voting in order to produce the classification results of the forest. Afterward, the samples that are not included in the bootstrap ones are used for the computation of the error produced by the RF. This error value, also called the out-of-bag (OOB) error, represents a performance measure of the tree-based algorithm (Belgiu and Drăguţ 2016). The OOB samples can also be regarded as the test set of a particular tree that is grown only by using the bootstrap data (Nguyen et al. 2013). In addition to that, the GINI index is a measure of the variable importance that is estimated by using this OOB data:

$$G_{(t)} = 1 - \sum_{k=1}^{Q} p^2(k|t) \qquad (1)$$

where t is related to a particular node, Q represents the number of classes and $p^2(k|t)$ is the estimated probability. Thus, the average decrease in this coefficient is computed for the features that are used for the split of particular nodes. The result shows how important these variables are in the entire RF.

Regarding the optimization of the performance of RF, there are several hyperparameters that play a key role in the training process (Belgiu and Drăguţ 2016). One of them is the number of levels which refers to the depth of each tree in the forest. Furthermore, the number of attributes used as the candidates for the best split of any node can also be tuned.

4.4.2 BNB

The Bernoulli Naïve Bayes is a discrete data Naïve Bayes classification algorithm, requiring binary input exclusively, meaning it can be only applied to data, consisting of boolean features. In case of using the algorithm on non-binary data, the “binarize” parameter of the model must be adjusted, meaning it must be assigned a certain float-threshold. The model in turn binarizes the handed input according to this threshold – values below or equal the threshold are assigned 0 and above it – 1. If the parameter is left empty or assigned 0, the algorithm performs upon the assumption, that the dataset is already binary. From all Naïve Bayes classifiers, the BNB is the one with most parameters to optimize. In addition to the “binarize” parameter, the algorithm has the parameters “alpha”, “fit_prior” and

"class_prior". "Alpha" is a smoothing parameter, which implies no smoothing if assigned 0 and maximal smoothing if assigned 1. The "fit_prior" determines whether the model should learn class prior possibilities or not. The default setting is a uniform prior. Finally, "class_prior" contains the prior probabilities of the classes. If no probabilities are handed, the priors are adjusted according to the data. The decision rule of the BNB looks as follows:

$$P(x_i|y) = P(i|y)x_i + \big(1 - P(i|y)\big)(1 - x_i) \qquad (2)$$

It is important pointing out that the BNB explicitly takes account of the absence of feature i for class y, since this is the sole difference to the Multinomial Naïve Bayes (McCallum and Nigam, 1998; Metsis et al., 2006).

4.5 Evaluation

The choice of the appropriate evaluation measure is essential for the performance assessment of the analytical models especially when these are trained on imbalanced data. Hence, Sokolova et al. (2006) suggest using the confusion matrix in order to gain knowledge of the prediction results in the first place. Thus, Table 2 shows how this matrix would look like in the context of a binary cyber-attack classification. The normal network activity is represents the negative class whereas the cyber-attack the positive one. Furthermore, Table 3 provides an overview of some of the most frequently used evaluation measures, their formulas and interpretation with respect to analyzing network intrusions.

In this context, recall can be regarded as some of the most important performance metrics as it reflects the ability of an IDS to capture the cyber-threats. Precision is a further highly relevant criteria as it shows how exact the predictions of an IDS are whenever an alarm for a pontential cyber-crime is raised. It is also essentail to examine the False Negative Rate (FNR) as it shows how many of the intrusions remain undetected. By contrast, using the overall accuracy for the evaluation of the results may lead to misleading conclusions especially when the class distribution of the data is imbalanced. Hence, Provost and Fawcett (1997) recommend using the Receiver operating characteristics (ROC) graph for the model evaluation, instead. For example, if the instances related to normal network activity represent 80% of the observations and the algorithms completely ignore the attacks in the prediction, then the accuracy would still be 0,8. Thus, using this measure may result in choosing a classifier that is completely useless in order to analyze cyber-crimes. By contrast, the ROC curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR). Therefore, it takes into account the ability of a prediction model to identify the cyber-attacks as well as the frequency with which the normal events are assigned to the positive class. Moreover, the Area under the ROC curve (AUC) represents a

numeric measure that signalizes how capable the machine learning algorithms are of distinguishing between the two classes (Ling et al. 2003).

4.6 Deployment

The last phase of CRISP-DM is the deployment of the best performing machine learning model. In the majority of the cases, the user is the one who carries out the necessary steps in this phase and not the data analyst. Therefore, it is of crucial importance to precisely determine the activities in advance that have to be performed in order to successfully integrate the model in the environment of the particular institution (Wirth and Hipp 2000).

5 Empirical Research

The accurate identification of cyber-attacks in advance is among the main goals of businesses that aim at minimizing security risks as much as possible. Therefore, this section describes the process of building an analytical model for the prediction of network threats.

5.1 Data Retrieval

The data is retrieved from the official website of the Data Mining and Knowledge Discovery and Data Mining competition (KDD Cup) https://www.kdd.org. The KDD 1999 Cup datasets were generated in order to provide researchers with a benchmark that can be used in the field of intrusion detection systems (IDS) (Olusola et al. 2010). A local area network (LAN) entvironment was created for this purpose and raw TCP/IP dump data was collected from it for the time period of nine weeks. The training data is built up by around 5 million observations that were acquiered from seven weeks of network traffic and the test data by 2 million records from 2 weeks. The LAN operated as an actual US. military network in which the target systems were blasted by various cyber-attacks. Hence, each of the records obtained from the simulation represents a connection that is either related to normal network activity or to a particular cyber-crime (Eldos et al., 2012; Hussain and Lalmuanawma, 2016; Kayacik et al., 2005; Protić, 2018).

In addition to that, both labeled and unlabeled datasets for network intrusion detection are made publically available on the official website of KDD Cup. In this regard, the intial aim of this research was to train the machine learning algorithms described in Section 4.4 on as many observations as possible for the binary classification of the records as legitimate or illegitimate network connections. However, the labeled training data of about 5 million samples could not be imported in Python for several hours using the computers at the PC-Pools in the Humboldt University in Berlin. Therefore, the available technical capacities played a major role in

choosing a subset that consists of approximately 10 percent of the labeled data for the analysis. The obtained subset is composed by 494.020 connection vectors and 42 variables including the target one.

The features can be divided into four main groups: basic, content, time-based and traffic-based attributes. The basic ones provide information about the network packets extracted from their header. Hence, this group of variables does not analyze the content of the units of network transmission. By contrast, the second group inspects the payload of the TCP packets. Thus, the features in it provide details about suspicious events like for instance the number of failed login attempts. Furthermore, the third group consists of variables computed based on the time interval of 2 seconds. In comparison to that, the estimation window of 100 connections in the fourth one implies that the respective attributes are suitable for the analysis of attacks that last longer than 2 seconds (Kayacik et al., 2005; Protić, 2018; Tavallaee et al., 2009).

5.2 Data Analysis

A deep understanding of the collected data is gained by describing its main characteristics as well as by using some visual elements like pie chart, treemap, countplot and stripplot. Data corrections are performed based on the results of the analysis.

5.2.1 Duplicated Observations

The retrieved data contains a huge amount of redundant records. Approximately 75 percent of the observations represent duplicated connection vectors. In this regard, Protić (2018) points out that this is due to the fact that the KDD cup 1999 data was collected from a virtual computer network. Hence the simulation itself of network traffic is highly likely to be the main reason for the duplicates.

In this context, it is important to highlight that training the analytical models on a dataset in which the majority of the observations are copies would increase the computational time despite the fact that no new information would flow in the learning process. Furthermore, Tavallaee et al. (2009) underline that the machine learning algorithms are expected to produce biased results. The classifiers would learn how to identify the most frequent instances and could possibly ignore the infrequent, distinct ones. Moreover, splitting the 10 percent KDD Cup dataset into train and test set wihtout removing the duplicates in advance could also lead to overfitting as both of these subsets would partially contain the same samples. In addition to this, overfitting refers to building a model with decreased general prediction capability that produces poor results when applied on unseen data (Sun et al. 2009). In this particular case the classifiers will identify the redundant records during the testing phase which may lead to misleadingly good

results in terms of particular evaluation criteria. Therefore, the redundant observations are removed in order to avoid the mentioned problems. The remaining data consists of 145.585 network traffic records.

5.2.2 Missing Values

Handling missing values is an essential part of the data preparation for the modeling phase as the majority of the machine learning algorithms do not tolerate lacking information. In this regard, the barplot in Figure 1 visualizes the patterns of missing data in the 10 percent KDD Cup dataset. Each of the bars represents a variable. The fact that every single bar is completely populated implies that there are no missings.

5.2.3 Class Distribution of the Target Variable

Next the distribution of the target variable is examined. Figure 2 shows the frequency of the respective classes. In addition to that, the categories related to normal network activity and Neptune attacks represent the majority of the instances. The Neptune attack is also known as a synchronization (SYN) flood attack. It is launched against a network host by sending a large amount of connection requests also called SYN packets from a spoofed Internet Protocol (IP) address. The TCP server which receives these requests sends back a response to the source in the form of a SYN / acknowledgement (ACK) packet and starts waiting for the source to respond with an ACK packet. In the context of computer networks, the term acknowledgement refers to a signal sent between devices in order to signalize that a particular message has been received. The attacker never responds to the TCP server. This exhausts the ressources of the receiving host as it keeps on waiting for a confirmation. As a result of this, the TCP server becomes unavailable to regular network connections (Haddadi and Beghdad 2018).

In addition to the class distribution of the target column, Aghdam and Kabiri (2016) summarize the 22 cyber-attacks into four main categories as shown in Table 4. Furthermore, the 3D pie chart in Figure 3 visualizes the class proportions of the transformed target variable. The category Denial of Service (DoS) is the most frequent one among the 4 cyber-threats as it includes the Neptune attacks, whereas Probing, R2L and U2R intrusions are associated with only 2.19 percent of the observations in the entire dataset. The line plot in Figure 4 shows the average duration of the legitimate and illegitimate connections. DoS attacks have an average time span close to 0 seconds. In this context, it is important to mention that the more requests a particular server receives from a DoS attacker in a very short time, the less time it will take until the server becomes partially or completely unaceessable. In comparison to this, Probing threats have the longest duration. This could possibly indicate that the attacker aims at

maintaining illegal access to the system of the victim for as long as possible in order to steal as much confidential information as possible. R2L intrusions are the second longest lasting attack on average in the KDD Cup data. This means that the attacker may spend a lot of time looking for vulnerable points in the security system of a network in order to gain illegal access later on. Finally, U2R threats last on average 500 seconds less than the R2L ones. This implies that the time window required for an U2R attacker to switch from a normal user to a root one is very narrow on average.

5.2.4 Encoding categorical Features

The distribution of the categorical features “protocol_type”, ”service” and “flag” is analysed in relation to the classes of the target column. For this purpose the U2R, R2L and Probing attacks are summarized into a new category called “other_attacks” due to their tiny proportions. The first part of the encoding of the categorical data is related to the reduction of the number of levels in the respective columns. This is essential especially in the case of high-cardinality attributes containing many distinct values. The second part is associated with dummy encoding performed by converting the remaining categories into new binary columns.

The feature “flag” contains a total of 11 unique values. The countplot in Figure 5 shows only 3 of them as the rest of the categories have very few occurrences. Thus the remaining labels are summarized into “other_flags”. SF refers to a normal status of the connection. This explains why over 90 percent of the regular network traffic records are associated with it. By contrast, the majority of the cyber-crimes, in particular the DoS intrusions, have the status S0 as it indicates that the target host received the connection requests, meaning the SYN packets, from the attacker who eventually never replied to the server. REJ is related to an error status and implies that a connection attempt is rejected by the network host. There are two times more illegitimate records than regular ones having this error status. Each of the four categories in Figure 5 is transformed to a new binary feature.

Regarding the distribution of “protocol_type”, the treemap in Figure 6 visualizes that TCP is the most frequent category, whereas UDP and ICMP represent very small proportions of the instances. In addition to this, the stripplot in Figure 7 depicts the three-way relationship between “protocol_type”, “serror_rate” and the target variable. If more than 50 percent of the connections established to the same host have a SYN error and use udp, then this always indicates that respective traffic records in the KDD Cup dataset always DoS attacks. In case tcp is used for the transportation of the data and the same SYN error-rate is oberserved, then the corresponding observations are highly likely related to one of the four attack types and not to a

normal event. Moreover, in the majority of the cases the regular network connections set up to the same target host are using tcp and have a SYN error rate lower than 50 percent.

A further analysis of the each of the protocol types is performed by examining the count of the corresponding services in relation to the class variable. The 3D count plot in Figure 8 shows that within most of the normal connections the data transferred over TCP is sent to a webserver which applies http in order to process the request of the client. It is a fact that HTTP uses TCP due to the reliability TCP offers. The plot is valuable, however, because it demonstrates that the dependence is also valid the other way around. This is explainable by the fact that HTTP is one of the most relevant protocols for accessing information or files stored online (e.g. on websites). Hence, it will always be used when possible, as it is in the case of TCP. By contrast, DoS attacks taking place over TCP mostly target virtual private networks. This implies that the hackers mainly aim at making VPNs inaccessible for other users by overloading the system. This can be justified by the fact that executing any other more complex attacks might come along with more difficulties due to the higher protection provided for VPNs. Regarding the rest of the cyber-attacks related to TCP, no particular pattern with respect to a certain destination service can be recognized.

As shown in Figure 9, similarly to the TCP case, DoS attacks executed over UDP mostly attempt affecting VPNs. However, the majority of the normal network activity cases is associated with a DNS protocol. It is a fact that DNS is mostly associated with UDP, since DNS uses UDP by default and only switches to TCP after the volume of the transmitted data has reached a particular threshold. What is interesting and requires further consideration is the fact the opposite is apparently also true - namely, that UDP is highly related to DNS. The technical advantage of DNS consists in enabling the user to reach an online destination without having to enter its IP-address. When working along with TCP, however, following disadvantages arise: longer latency and higher costs for both clients and servers due to the necessity of establishing a connection when using TCP (Včelák 2017). Since DNS can only be found in UDP's five most used services and in the same time it corresponds to solely 5000 of all 145 000 observations, one can draw the conclusion, that the use of DNS is indeed preferred, but only as long as it does not entail further complications like the ones related to TCP. This means that the costs resulting from a DNS using TCP outweigh the benefits of DNS itself.

Finally, Figure 10 plots the counts of the top five services used by an ICMP. The highest count in the context of DoS attacks (and in general within ICMP) belongs to the echo reply service. Logically, this makes sense, since multiple DoS attacks – Ping of Death, Smurf etc. – consist in a hacker overflooding a system with echo requests until it eventually crashes, because it

cannot keep up with the echo replies (Sandeep 2014). This implies that a system sends out numerous echo replies when being under a DoS attack, which explains the results from the plot. The second highest count is the one of the echo request service, which is strongly related to attacks other than DoS attacks. This is explainable by the fact that most probing attacks (second biggest count after DoS attacks) – like e.g IP Address Sweep – occur when a source host sends ICMP packets to the destination host. As a result of the performed analysis, the categories different from HTTP, private service, SMTP, DNS and ECR in the variable service are summarized into "other_services". Afterward, each of the levels is converted to a new binary feature.

5.2.5 Feature Selection

To begin with, the features "num_outbound_cmds" and "is_host_login" contain only the value "0". This means that they cannot contribute to the performance of the analytical models in any possible way. Hence, they are filtered out.

Furthermore, a correlation filter is applied in order to eliminate the features that have a strong correlation relationship with each other. Sensitivity analysis is performed in order to find the best threshold for the filter. This kind of analysis involves identifying the values and the parameters that improve the performance of the analytical models (Pannell 1997). The filter is applied once the categories in the class variable "DoS" and "other-attacks" are summarized in "cyber-attacks" in order to create a binary classification problem. Furthermore, the features created as a result of encoding of the categorical data are excluded from the correlation filter. These variables correlate a lot with each other as they initially represented the categories in the columns "protocol_type", "flag" and "service". Therefore, the strong correlation for instance between "protocol_type_tcp", "protocol_type_udp" and "protocol_type _icmp" results from the data preprocessing described in Section 5.2.4. Furthermore, as already mentioned in Section 4.4.2 BNB requires binary variables as input for the training process. Therefore, the features in the KDD Cup 1999 dataset that do not contain boolean values are binarized. The mean of each column is chosen as the threshold for the binarization. This means that values exceeding the mean in a particular variable are replaced with 1s and the ones equal to or below the mean with 0s. In this context, it is important to mention that no binarization is performed for RF as tree-based algorithms handle not only binary data, but also other types of features, for instance continuous ones. The DR is used as the main evaluation mertric in order to determine the best results. In addition to this, the corresponding values of the precision score, FNR, AUC and FPR are also examined.

Figure 11 shows that RF achieves the best results in terms of all evaluation criteria when a correlation threshold of 0.8 is applied. The respective values of the DR, precision and AUC are the maximal ones and the score of the FNR as well as FPR is the minimal one. Therefore, all features that correlate more than 80 percent with each other are eliminated in order to obtain the optimal subset for the modeling of the hyperparameters of RF in the next phase.

Figures 12 and 13 show the results that BNB yields with and without binarization of the input data. The algorithm achieves without any transformation the highest DR when a correlation threshold of 0.9 is applied. In comparison to that, a threshold of 0.7 leads to the best result in terms of the DR when binarization is performed. In addition to this, Table 5 summarizes the results in relation to the maximal DR achieved in both cases. BNB proves to perform better when the feature space is binarized. Hence, the columns containing more than 70 percent redundant information are removed and the rest of the variables are converted to boolean ones for the further analysis with BNB.

5.3 Modeling and Evaluation

In the last phase, particular parameters of RF and BNB are optimized by performing sensitivity analysis. A grid of parameters is defined for each of the two algorithms and all possible combinations are tested on the analytical models.

In the case of RF, different values for four hyperparameters are iteratively tested. Table 6 provides an overview of the respective grid and the meaning of each of the parameters in it. Regarding BNB, two hyperparameters are optimized as shown in Table 7. Figures 14 and 15 provide the final results of the modeling phase. The main drawback of RF is the computational cost as a total 16 iterations lasted 5 times longer than 1.620 iterations performed with BNB. However, the maximal DR achieved by RF is higher than the one by BNB. This implies that the tree-based algorithm would outperform BNB in identifying the cyber-attacks once applied on unseen data. Both analytical models achieve a precision higher than 99.0% which indicates that less than 1% of all predictions as a cyber-threat turn out to be false. By contrast, the FNR of RF is 4,89% lower than the one of BNB. This means that less attacks remain undetected by RF than by BNB. In addition to this, the higher AUC value of RF shows that the tree-based model is also more capable of differenciating between the positive and the negative class than the Naïve Bayes algorithm. Furthermore, RF performs better than BNB also in terms of the FPR. Therefore, the analysis of each of the mentioned evaluation criteria except the computational time clearly indicates that RF should be chosen over BNB and deployed for the prediction of cyber-attacks.

5.4 Discussion and Analysis of the Results

The network intrusion analysis conducted in this research provides a tree-based predictive model for the identification of cyber-crimes. The initial result of the dimensionality reduction performed by iteratively applying different thresholds for the correlation filter shows that RF yields better results than BNB in terms of all five evaluation metrics: DR, precision, FNR, AUC and FPR. In this context, binarizing the feature space before the training process represents necessary preprocessing step in order to improve the performance of BNB in case the input data does not consist of boolean variables. Furthermore, both of the machine learning algorithms predict the cyber-crimes more accurately once some of the features holding redundant information are eliminated. This underlines the necessity to identify and remove variables that have a strong correlation relationship between each other. The optimization of the hyperparameters lead to results similar to the ones yielded from the feature selection. Even after the Naïve Bayes is trained in 1620 iterations in order to find the optimal set of parameters, RF still outperforms BNB. As the computational cost of the training process of BNB is significantly lower than the one of RF, the tree-based algorithm proves to identify the network intrusions in a more effective, but less efficient way than the Naïve Bayes model.

6 Conlusion

This research aimed at building an analytical model that identifies as many cyber-attacks as possible. For this purpose, the methodology of CRISP-DM was followed. The retrieval of relevant data was the first big challenge during the project as almost none of the institutions make their own network traffic data publically availably due to data security reasons. In addition to that, the 10 percent KDD Cup dataset obtained from the official website of KDD consisted of features that provide not only basic information about the network packets but also details about their content. Furthermore, one part of the features was computed based on the time-period of 2-seconds whereas another one based on the estimation window of 100 connections. In this context, analyzing the collected data in order to find hidden patterns and quality problems in it was the most time-consuming phase. The obtained data turned out to have no missing values. However, the duplicated observations represented the majority of the samples which lead to a reduction of approximately 75 percent of the traffic records. Moreover, the variables having a strong correlation relationship with each other were eliminated. The results of the dimensionality reduction as well as of the modeling phase showed that RF achieves better results in terms of all considered evaluation criteria.

Therefore, the tree-based model should be deployed on unseen data. It is advisable to document each of the actions that need to be executed in a very detailed way in order to successfully identify the cyber-attacks in advance. Furthermore, it is recommended to deploy the model repeatedly and to monitor its performance precisely.

7 Reference List

Abdullah, M., Alshannaq, A., Balamash, A. & Almabdy, S. (2018). Enhanced Intrusion Detection System using Feature Selection Method and Ensemble Learning Algorithms. *International Journal of Computer Science and Information Security (IJCSIS), 16 (2)*, 48–55.

Abdurrazaq, M. N. K., Trilaksono, B. R. & Rahardjo, B. (2015). DIDS Using Cooperative Agents Based on Ant Colony Clustering. *Journal of ICT Research and Applications, 8 (3)*, 213-233.

Aggarwal, M. & Amrita, M. (2013). Fusion of Statistic, Data Mining and Genetic Algorithm for feature selection in Intrusion Detection. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 2 (5)*, 1725–1731.

Aghdam, M. H. & Kabiri, P. (2016). Feature Selection for Intrusion Detection System Using Ant Colony Optimization. *International Journal of Network Security, 18 (3)*, 420–432.

Ahmed Elnaggar. (2015). *TCP Vs. UDP*. Technical Report.

Aissa, N. B. & Guerroumi, M. (2016). Semi-supervised Statistical Approach for Network Anomaly Detection. *Procedia Computer Science 83,* 1090–1095.

Altwaijry, H. & Algarny, S. (2012). Bayesian based intrusion detection system. *Journal of King Saud University - Computer and Information Sciences, 24 (1)*, 1–6.

Anil Kumar, D. & Ravi, V. (2008). Predicting credit card customer churn in banks using data mining. *International Journal of Data Analysis Techniques and Strategies, 1 (1)*, 4–28.

Arkin, O. (2000). *ICMP Usage in Scanning Or Understanding some of the ICMP Protocol's Hazards*. Technical Report.

[Artificial Intelligence - All in One]. (2017). *Lecture 15.5 — Anomaly Detection | Anomaly Detection Vs Supervised Learning — [Andrew Ng]* [Video File]. Retrieved from https://www.youtube.com/watch?v=lCipWj-Cets. Accessed on: 05.02.2019.

Ashraf, N., Ahmad, W. & Ashraf, R. (2018). A Comparative Study of Data Mining Algorithms for High Detection Rate in Intrusion Detection System. *Annals of Emerging Technologies in Computing (AETiC), 2 (1)*, 48–57.

Bajaj, K. & Arora, A. (2013). Improving the Intrusion Detection using Discriminative Machine Learning Approach and Improve the Time Complexity by Data Mining Feature Selection Methods. *International Journal of Computer Applications, 76 (1)*, 5–11.

Basak, A. & Thaseen, S. (2014). *A Hybrid Intrusion Detection model using PCA, K-means clustering and Bayes' Theorem*. Report.

Batista, G. E., Prati, R. C. & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM Sigkdd Explorations Newsletter, 6 (1)*, 20–29.

Belgiu, M. & Drăguţ, L. (2016). Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing, 114,* 24–31.

Bolon-Canedo, V., Sanchez-Marono, N. & Alonso-Betanzos, A. (2009). A combination of discretization and filter methods for improving classification performance in KDD Cup 99 dataset. In *2009 International Joint Conference,* 359–366.

Breiman, L. (2001). Random forests. *Machine learning, 45 (1)*, 5–32.

Brownlee, J. (2017). How to Handle Missing Data with Python. https://machinelearningmastery.com/handle-missing-data-python/. Accessed on: 28.02.2019.

Brownlee, J. (2018). How to Use Statistics to Identify Outliers in Data. https://machinelearningmastery.com/how-to-use-statistics-to-identify-outliers-in-data/. Accessed on: 28. Februar 2019.

Burke, J. (2017). Was ist der Unterschied zwischen dem TCP/IP- und dem OSI-Modell? https://www.computerweekly.com/de/antwort/Was-ist-der-Unterschied-zwischen-dem-TCP-IP-und-dem-OSI-Modell. Accessed on: 19.03.2019.

Chandra, P., Lilhore, U. K. & Agrawal, N. (2017). NETWORK INTRUSION DETECTION SYSTEM BASED ON MODIFIED RANDOM FOREST CLASSIFIERS FOR KDD CUP-99 AND NSL-KDD DATASET. *International Research Journal of Engineering and Technology (IRJET), 4 (8)*, 786–791.

Chandradevan, R. (2017). Random Forest Learning-Essential Understanding. https://towardsdatascience.com/random-forest-learning-essential-understanding-1ca856a963cb. Accessed on: 10.01.2019.

Chaudhari, R. R. & Patil, S. P. (2017). INTRUSION DETECTION SYSTEM: CLASSIFICATION, TECHNIQUES AND DATASETS TO IMPLEMENT. *International Research Journal of Engineering and Technology (IRJET), 4 (2)*, 1860–1866.

Chopra, S. (2018). How to Handle Missing Data in Machine Learning: 5 Techniques. https://dev.acquia.com/blog/how-to-handle-missing-data-in-machine-learning-5-techniques/09/07/2018/19651. Accessed on: 09.01.2019.

Choudhary, P. (2017). Introduction to Anomaly Detection. https://www.datascience.com/blog/python-anomaly-detection. Accessed on: 02.02.2019.

Dahiya, P. & Srivastava, D. K. (2018). Network Intrusion Detection in Big Dataset Using Spark. *Procedia Computer Science, 132,* 253–262.

Dhanabal, L. & Shantharajah, S. P. (2015). A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms. *International Journal of Advanced Research in Computer and Communication Engineering, 4 (6)*, 446–552.

Duque, S. & Omar, M. N. b. (2015). Using Data Mining Algorithms for Developing a Model for Intrusion Detection System (IDS). *Procedia Computer Science, 61,* 46–51.

Eldos, T., Siddiqui, M. K. & Kanan, A. (2012). On the KDD'99 Dataset: Statistical Analysis for Feature Selection. *Journal of Data Mining and Knowledge Discovery, 3 (3)*, 88–90.

Elekar, K. S. & Waghmare, M. M. (2014). Study of Tree Base Data Mining Algorithms for Network Intrusion Detection. *International Journal on Recent and Innovation Trends in Computing and Communication, 2* (*10)*, 3253–3257.

Emani, B. B. R., Sonthi, V. K. & Maddipati, S. S. (2016). A Novel Filtered Classification Algorithm for Network intrusion detection. *International Research Journal of Engineering and Technology (IRJET), 3 (6)*, 2750–2753.

Farnaaz, N. & Jabbar, M. A. (2016). Random Forest Modeling for Network Intrusion Detection System. *Procedia Computer Science, 89,* 213–217.

Fruhlinger, J. (2018). What is a cyber attack? Recent examples show disturbing trends. https:// www.csoonline.com/article/3237324/what-is-a-cyber-attack-recent-examples-show-disturbing-trends.html. Accessed on: 20.03.2019.

Haddadi, M. & Beghdad, R. (2018). DoS-DDoS: TAXONOMIES OF ATTACKS, COUNTERMEASURES, AND WELL-KNOWN DEFENSE MECHANISMS IN CLOUD ENVIRONMENT. *EDPACS, 57 (5)*, 1–26.

Hall, M. (2000). Correlation-based feature selection for discrete and numeric class machine learning. In *Proceedings of the Seventeenth International Conference on Machine Learning,*

359–366.

Hussain, J. & Lalmuanawma, S. (2016). Feature Analysis, Evaluation and Comparisons of Classification Algorithms Based on Noisy Intrusion Dataset. *Procedia Computer Science, 92,* 188–198.

Jain, Y. & Jain, P. (2017). Improving the Performance of Intrusion Detection System by Removing the Count Attribute from KDD Cup 1999 Data. *International Journal for Scientific Research & Development, 5 (8)*, 180–183.

Kayacik, H. G., Zincir-Heywood, A. N. & Heywood, M. I. (2005). Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets. In *Proceedings of the third annual conference on privacy, security and trust*.

Keshta, I. M. (2018). Intelligent Intrusion Detection System Based on MLP, RBF and SVM Classification Algorithms: A Comparative Study. *International Journal of Computer Science and Information Security (IJCSIS), 16 (5)*, 23–30.

Khan, A. & Khan, S. (2008). Two Level Anomaly Detection Classifier. In *2008 International Conference on Computer and Electrical Engineering,* 65–69.

Kotsiantis, S., Kanellopoulos, D. & Pintelas, P. (2006). Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering, 30 (1)*, 25-36.

Kwak, S. K. & Kim, J. H. (2017). Statistical data preparation: management of missing values and outliers. *Korean journal of anesthesiology, 70 (4)*, 407–411.

Langley, P. & Simon, H. A. (1995). Applications of machine learning and rule induction. *Communications of the ACM*, *38 (11)*, 54-64.

Lawrence, R. L. & Wright, A. (2001). Rule-based classification systems using classification and regression tree (CART) analysis. *Photogrammetric engineering and remote sensing, 67 (10)*, 1137-1142.

Ling, C. X., Huang, J. & Zhang, H. (2003). AUC: a statistically consistent and more discriminating measure than accuracy. In *Proceedings of 18th International Conference on*

Artificial Intelligence, 329-341.

Luber, S. (2018a). Was ist ein VPN? https://www.ip-insider.de/was-ist-ein-vpn-a-625331/. Accessed on: 03.03.2018.

Luber, S. (2018b). Was ist HTTP (Hypertext Transfer Protocol)? https://www.ip-insider.de/was-ist-http-hypertext-transfer-protocol-a-691181/. Accessed on: 04.03.2019.

Luber, S. (2018c). Was ist SMTP (Simple Mail Transfer Protocol)? https://www.ip-insider.de/was-ist-smtp-simple-mail-transfer-protocol-a-691215/. Accessed on: 02.03.2019.

Mazini, M., Shirazi, B. & Mahdavi, I. (2018). Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms. *Journal of King Saud University - Computer and Information Sciences,* 1-13.

McCallum, A. & Nigam, K. (1998). A comparison of event models for naive Bayes text classification. In *Proc. AAAI/ICML-98 Workshop on Learning for Text Categorization,* 41-48.

Metsis, V., Androutsopoulos, I. & Paliouras, G. (2006). Spam filtering with naive Bayes – Which naive Bayes? In *3rd Conf. on Email and Anti-Spam (CEAS)*.

Naser, D. (2017). Supervised vs. Unsupervised Learning. https://content.nexosis.com/blog/supervised-vs-unsupervised-learning. Accessed on: 01.02.2019.

Nguyen, C., Wang, Y. & Nguyen, H. N. (2013). Random forest classifier combined with feature selection for breast cancer diagnosis and prognostic. *Journal of Biomedical Science and Engineering, 6 (5)*, 551–560.

Olusola, A. A., Oladele, A. S. & Abosede, D. O. (2010). Analysis of KDD'99 intrusion detection dataset for selection of relevance features. In *Proceedings of the World Congress on Engineering and Computer Science,* 20–22.

Onik, A. R., Haq, N. F., Alam, L. & Mamun, T. I. (2015). An Analytical Comparison on Filter Feature Extraction Method in Data Mining using J48 Classifier. *International Journal of Computer Applications, 124 (13)*, 1–8.

Osborne, J. W. & Overbay, A. (2004). The power of outliers (and why researchers should always check for them). *Practical assessment, research & evaluation, 9 (6)*, 1–12.

Pannell, D. J. (1997). Sensitivity analysis of normative economic models: theoretical framework and practical strategies. *Agricultural economics, 16 (2)*, 139–152.

Prasad, U. D. & Madhavi, S. (2012). Prediction of churn behavior of bank customers using data mining tools. *Business Intelligence Journal, 5 (1)*, 96–101.

Priyadarsini, P. I., Anuradha, C. & Murty, P. S .R. Chandra. (2017). Fuzzy Based Feature Selection for Intrusion Detection System. *International Journal of Engineering Research in Computer Science and Engineering, 4 (11)*, 42–49.

Protić, D. (2018). Review of KDD Cup '99, NSL-KDD and Kyoto 2006+ datasets. *Vojnotehnicki glasnik, 66 (3)*, 580–596.

Provost, F. & Fawcett, T. (Hrsg.). (1997). Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining,* 43-48.

Ray, S. (2016). A Comprehensive Guide to Data Exploration. https://www.analyticsvidhya.com/blog/2016/01/guide-data-exploration/. Accessed on: 29.12.2018.

Rohrer, B. (2015). Methods for handling missing values. https://gallery.azure.ai/Experiment/Methods-for-handling-missing-values-1. Accessed on: 25.01.2019.

Rouse, M. (2014). HTTP (Hypertext Transfer Protocol). https://whatis.techtarget.com/de/definition/HTTP-Hypertext-Transfer-Protocol. Accessed on: 05.03.2019.

Sahu, P., Saxena, A. & Singh, K. (2017). Augment Method for Intrusion Detection around KDD Cup 99 Dataset. *International Research Journal of Engineering and Technology (IRJET), 4 (6)*, 2424–2431.

Sandeep, R. (2014). A Study of DOS & DDOS – Smurf Attack and Preventive Measures. *International Journal of Computer Science and Information Technology Research, 2 (4)*, 312–317.

Santoyo, S. (2017). A Brief Overview of Outlier Detection Techniques. https://towardsdatascience.com/a-brief-overview-of-outlier-detection-techniques-1e0b2c19e561. Accessed on: 28.12.2018.

Saraswati, A., Hagenbuchner, M. & Zhou, Z. Q. (2016). High Resolution SOM Approach to Improving Anomaly Detection in Intrusion Detection Systems. In *Australasian Joint Conference on Artificial Intelligence 2016: Advances in Artificial Intelligence,* 191-199.

Shearer, C. (2000). The CRISP-DM Model: The New Blueprint for Data Mining. *Journal of Data Warehousing, 5 (4)*, 13–22.

Shrivas, A. K. & Mishra, P. K. (2016). Intrusion Detection System for Classification of Attacks with Cross Validation. *International Journal of Engineering Science Invention, 5 (9)*, 21–24.

Siddiqui, M. K. & Naahid, S. (2013). Analysis of KDD CUP 99 Dataset using Clustering based Data Mining. *International Journal of Database Theory and Application, 6 (5)*, 23–34.

Sokolova, M., Japkowicz, N. & Szpakowicz, S. (2006). Beyond accuracy, F-score and ROC: A family of discriminant measures for performance evaluation. In *Proceedings of the ACS Australian joint conference on artificial intelligence,* 1015-1021.

Srocke, D. (2018). Was ist DNS (Domain Name System)? https://www.ip-insider.de/was-ist-dns-domain-name-system-a-579256/. Accessed on: 02.03.2019.

Strotmann, C. (2019). DNS flag day: Initiative gegen veraltete DNS-Server und Middleboxen. https://www.heise.de/newsticker/meldung/DNS-flag-day-Initiative-gegen-veraltete-DNS-Server-und-Middleboxen-4289779.html. Accessed on: 09.03.2019.

Sun, Y., Wong, A. K. & Kamel, M. S. (2009). Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence, 23 (4)*, 687–719.

Tavallaee, M., Bagheri, E., Lu, W. & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications,* 1–6.

Tiwari, V. N., Rathore, S. & Patidar, K. (2016). Enhanced Method for Intrusion Detection over KDD Cup 99 Dataset. *International Journal of Current Trends in Engineering & Technology, 2 (2)*, 218–224.

Upadhyay, S. & Singh, R. R. (2015). Comparative Analysis based Classification of KDD'99 Intrusion Dataset. *International Journal of Computer Science and Information Security, 13 (5)*, 14–20.

Včelák, J. (2017). DNS OVER TCP AS SEEN FROM THE AUTHORITATIVE SERVER. https://ns1.com/blog/dns-over-tcp-as-seen-from-the-authoritative-server. Accessed on: 15.12.2018.

Verma, A. & Ranga, V. (2018). Statistical analysis of CIDDS-001 dataset for Network Intrusion Detection Systems using Distance-based Machine Learning. *Procedia Computer Science, 125,* 709–716.

Wirth, R. & Hipp, J. (2000). CRISP-DM: Towards a standard process model for data mining. In *Proceedings of the Fourth International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, 29-39.

8 Appendix

8.1 Table Appendix

Dataset	Problem Domain	Machine Learning Algorithms	Source
KDD-Cup 1999	Anomaly Detection	Bayesian Filter	(Altwaijry and Algarny 2012)
		High Resolution Self-Organizing Map	(Saraswati et al. 2016)
	Classification Anomaly Detection	Ant Colony Clustering	(Abdurrazaq et al. 2015)
	Classification	Naive Bayes, C4.5 – Decision Tree based Classifier	(Bolon-Canedo et al. 2009)
	Anomaly Detection	K-Means Clustering	(Siddiqui and Naahid 2013)
	Classification	C4.5 - Decision Tree based Classifier, Nayve Bayes	(Khan and Khan 2008)
	Classification	J48 Decision Tree, Hoeffding Tree, Random Forest, Random Tree, REP Tree	(Elekar and Waghmare 2014)
	Classification	J48 Decision Tree, Naïve Bayes, Random Forest	(Tiwari et al. 2016)
	Classification	Radial Basis Functions, Multilayer Perceptrons, Support Vector Machines	(Keshta 2018)
	Anomaly Detection	K-Means Clustering	(Jain and Jain 2017)
	Classification	Support Vector Machines	(Priyadarsini et al. 2017)
	Classification	Naïve Bayes	(Aggarwal and Amrita 2013)
	Classification	Naïve Bayes, J48 Decision Tree, J48 Craft, Bayes Net, Random Forest	(Upadhyay and Singh 2015)
	Classification	Naïve Bayes, J48 Decision Tree, Random Forest	(Sahu et al. 2017)
	Classification	Decision Trees	(Emani et al. 2016)

NSL-KDD	Classification	Random Forest	(Farnaaz and Jabbar 2016)
		K-Means Clustering	(Duque and Omar 2015)
		J48 Decision Tree, SVM, Naive Bayes	(Dhanabal and Shantharajah 2015)
		Nayve Bayes, K-Means Clustering	(Basak and Thaseen 2014)
		J48 Decision Tree	(Onik et al. 2015)
		Combination of C4.5 classifier, Classification Tree and Regression Tree	(Shrivas and Mishra 2016)
		J48 Decision Tree, Naïve Bayes, Naïve Bayes Tree, Multi-Layer Perception, Lib Support Vector Machines, Regression Tree	(Bajaj and Arora 2013)
		J48 Decision Tree, Random Forest, Partial Decision List, Ensemble Classifier	(Abdullah et al. 2018)
		Naïve Bayes, J48 Decision Tree, Random Forest	(Ashraf et al. 2018)
NSL-KDD KDD-Cup 1999	Classification	Random Forest, Modified Random Forest (Combination of unpruned classifiers and Regression Tree), J48 Decision Tree, Regression Tree, Naïve Bayes	(Chandra et al. 2017)
NSL-KDD, ISCXIDS2012	Anomaly Detection	AdaBoost	(Mazini et al. 2018)
NSL-KDD, Kyoto 2006+		Discriminant Function Regularization Iterative Anomlay Repartition	(Aissa and Guerroumi 2016)
CIDDS-001	Classification	Distance-based Machine Learning	(Verma and Ranga 2018)
UNSW NB-15	Classification Anomaly Detection	Naive Bayes, REP Tree, Random Tree, Random Forest, Random	(Dahiya and Srivastava 2018)

		Committee, Bagging, Randomizable Filtered Classifier	
KDD-Cup 1999, NSL-KDD, GureKDD	Classification Anomaly Detection	Support Vector Machines, Decision Tree, Naïve Bayes K-Means Clustering, K-Medoids Clustering	(Chaudhari and Patil 2017)

Table 1: Related Research

		Prediction	
		Cyber-Attacks	**Normal Network Activity**
Actual	**Cyber-Attacks**	True Positives (TP)	False Negatives (FN)
	Normal Network Activity	False Positives (FP)	True Negative (TN)

Table 2: Confusion Matrix

Measure	**Formula**	**Interpretation**
Overall Accuracy	(TP + TN) / (TP + FP + FN + TN)	How many of the observations in the entire dataset are classified correctly?
Error Rate	(FP + FN) / (TP + TN + FN + FP)	How many of the predictions are false?
False Positive Rate / False Alarm rate	FP / (FP + TN)	How many of the instances that truly represent normal network activity are falsely labeled as a cyber-attack?
False Negative Rate	FN / (FN + TP)	How many of the actual cyber-attacks are labeled as regular network traffic?

Recall / TP Rate/ Detection Rate (DR)	TP / (TP + FN)	How many of the actual cyber-attacks are identified by the classifier?
Precision	TP / (TP + FP)	How many of the records labeled as a cyber-attack are truly one?

Table 3: Evaluation Metrics

Categories	**Types**
Denial of Service	back, land, neptune, pod, smurf, teardrop
Probing	satan, ipsweep, nmap, portsweep
User to Root	buffer_overflow, loadmodule, perl, rootkit
Remote to Local	ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, warezmaster

Table 4: Types of Cyber-Attacks

	Max DR	**Correlation-threshold**	**Corresponding evaluation metrics**			
			Precision	**FNR**	**AUC**	**FPR**
Without Binarization	93.88%	0.9	99.38%	6.12%	96.75%	0.39%
With Binarization	94.66%	0.7	99.82%	5.34%	97.27%	0.11%

Table 5: Summary Correlation Filter BNB

Hyperparameter	**Meaning**	**Tested Values**
n_estimators	Number of trees in the entire forest	500, 1.000
max_depth	Maximal number of levels of each tree in the forest	60, 80
min_samples_leaf	Minimal number of instances in the leaf node	1, 3
min_samples_split	Minimal number of samples in a node before a split is performed	5, 10

Table 6: Tested Hyperparameters RF

Hyperparameter	**Meaning**	**Tested Values**
alpha	Laplace / Additive Smoothing of the distribution of the input data	From 0.1 to 1.0, step size=0.01

class_prior	The prior probabilities of the classes	For each of the classes from 0.1 to 0.9, step size: 0.05

Table 7: Tested Hyperparameters BNB

8.2 Figure Appendix

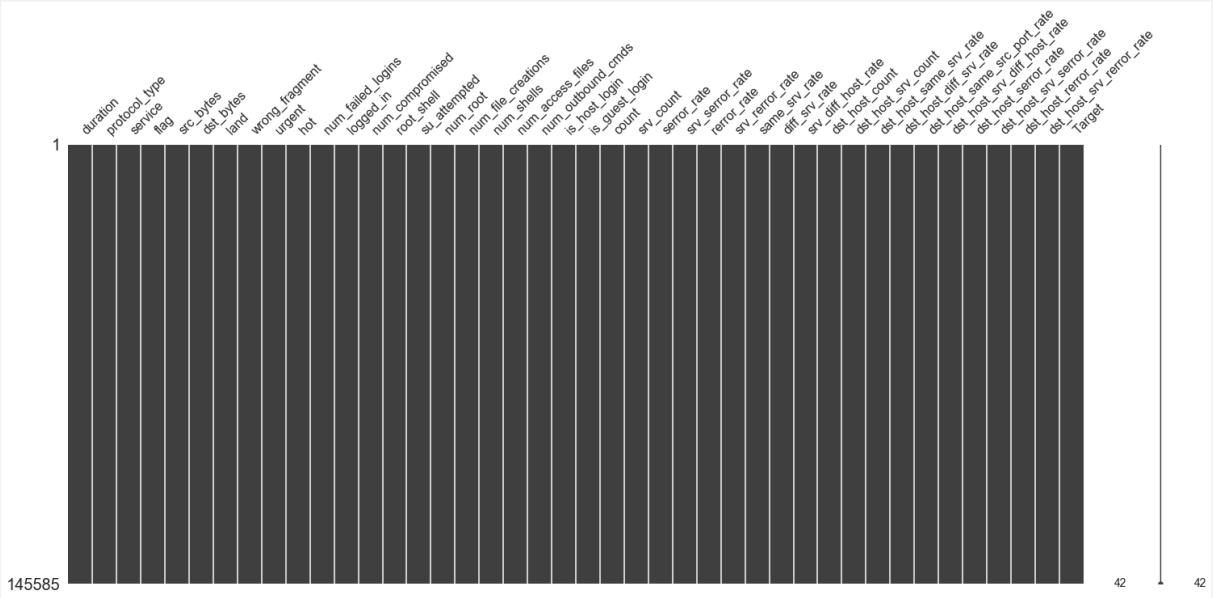


Figure 1: Missing Values

```
In [12]: print(values)
normal.              87831
neptune.             51820
back.                  968
teardrop.              918
satan.                 906
warezclient.           893
ipsweep.               651
smurf.                 641
portsweep.             416
pod.                   206
nmap.                  158
guess_passwd.           53
buffer_overflow.        30
warezmaster.            20
land.                   19
imap.                   12
rootkit.                10
loadmodule.              9
ftp_write.               8
multihop.                7
phf.                     4
perl.                    3
spy.                     2
```

Figure 2: Values Count Target Variable

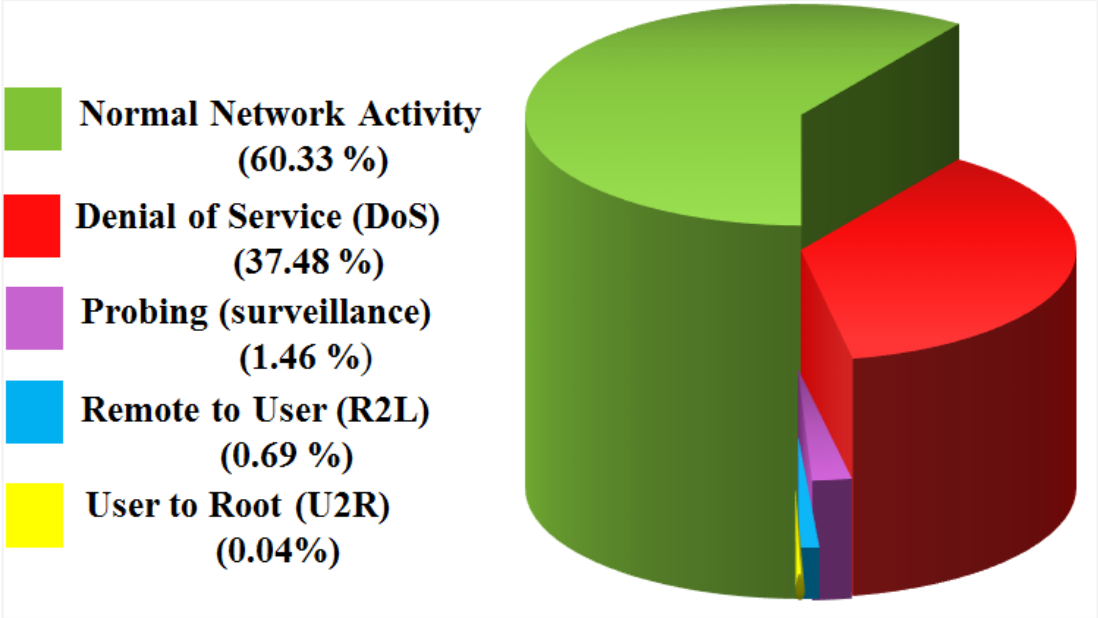


Figure 3: Categories of the transformed Target Variable

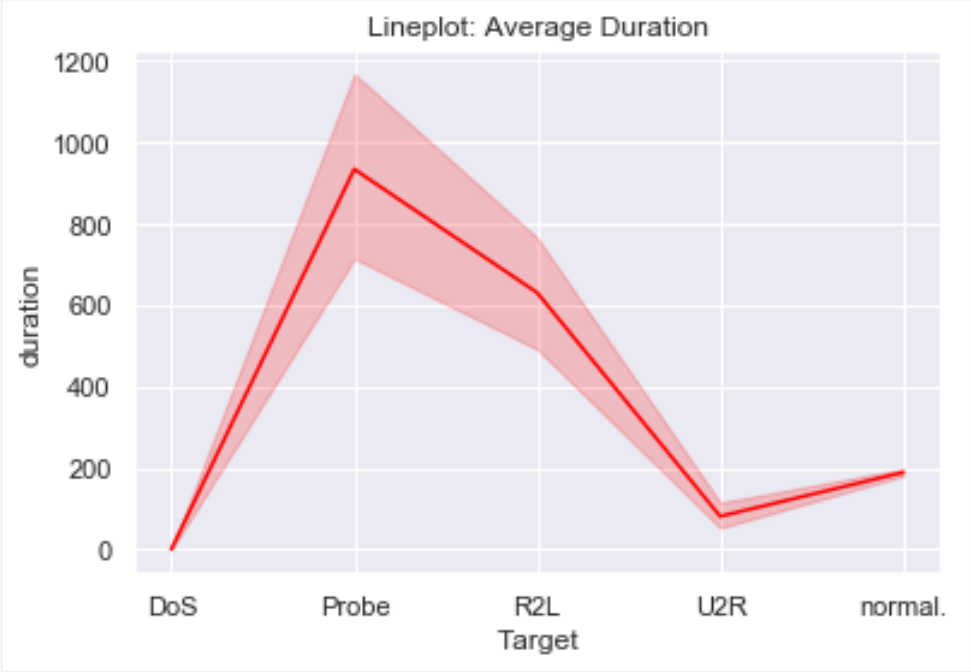


Figure 4: Average Duration Cyber-Attacks

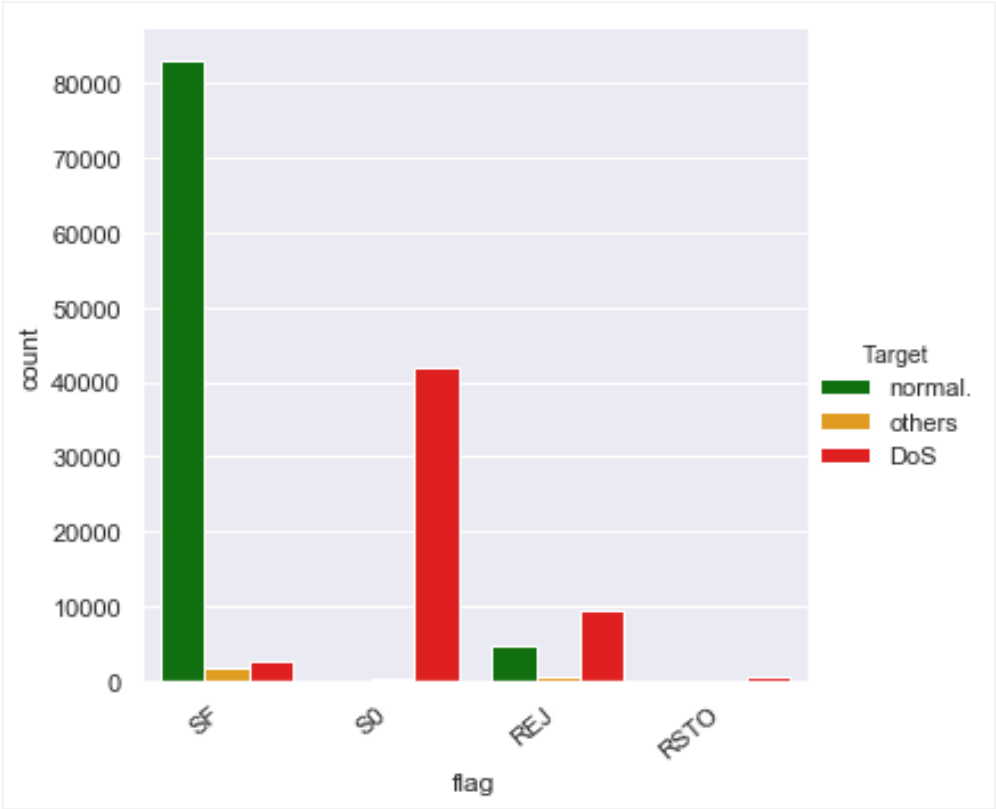


Figure 5: Countplot Flag

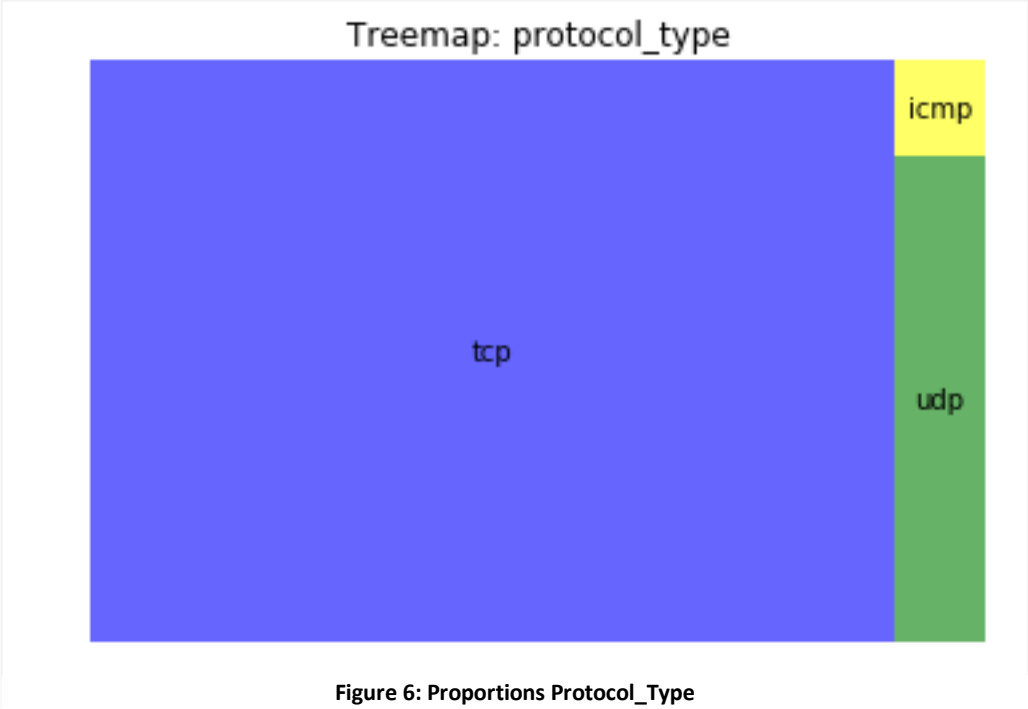


Figure 6: Proportions Protocol_Type

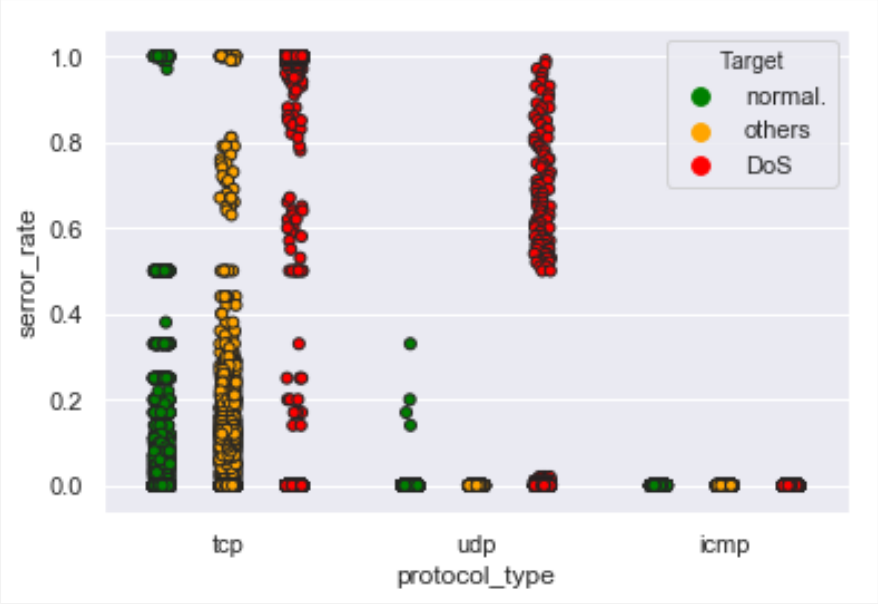


Figure 7: Stripplot serror_rate, protocol_type and Target

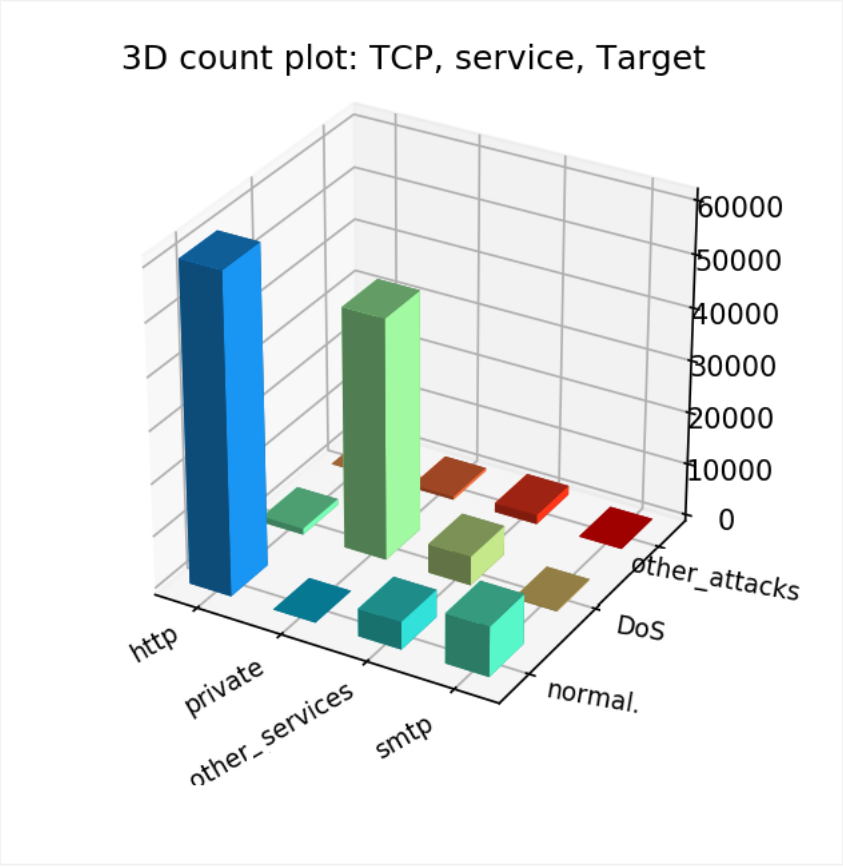


Figure 8: 3D Count plot TCP, service and Target

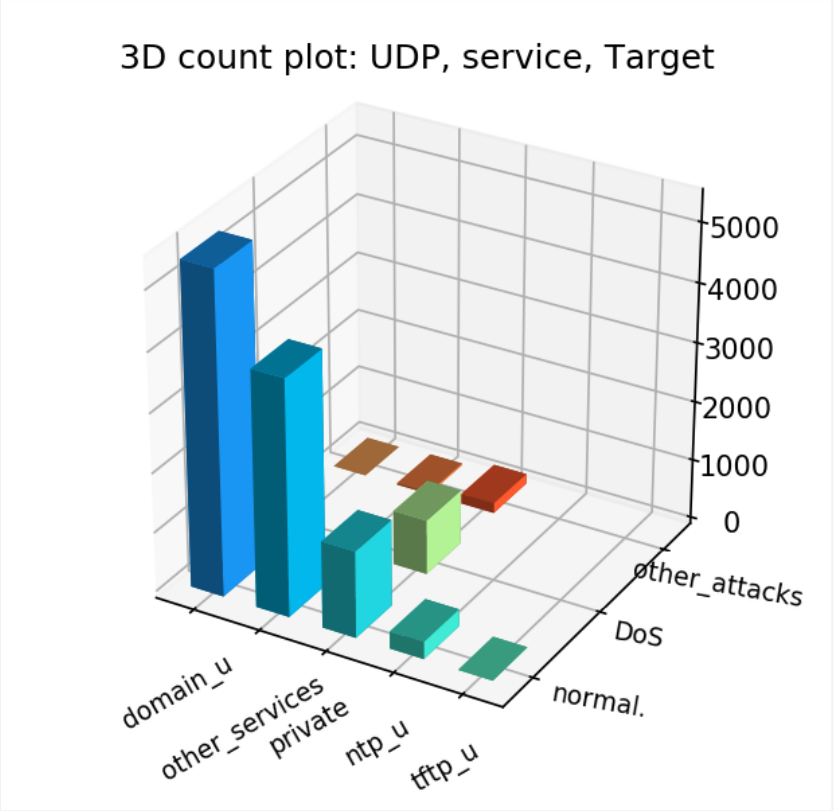


Figure 9: 3D Count plot UDP, service and Target

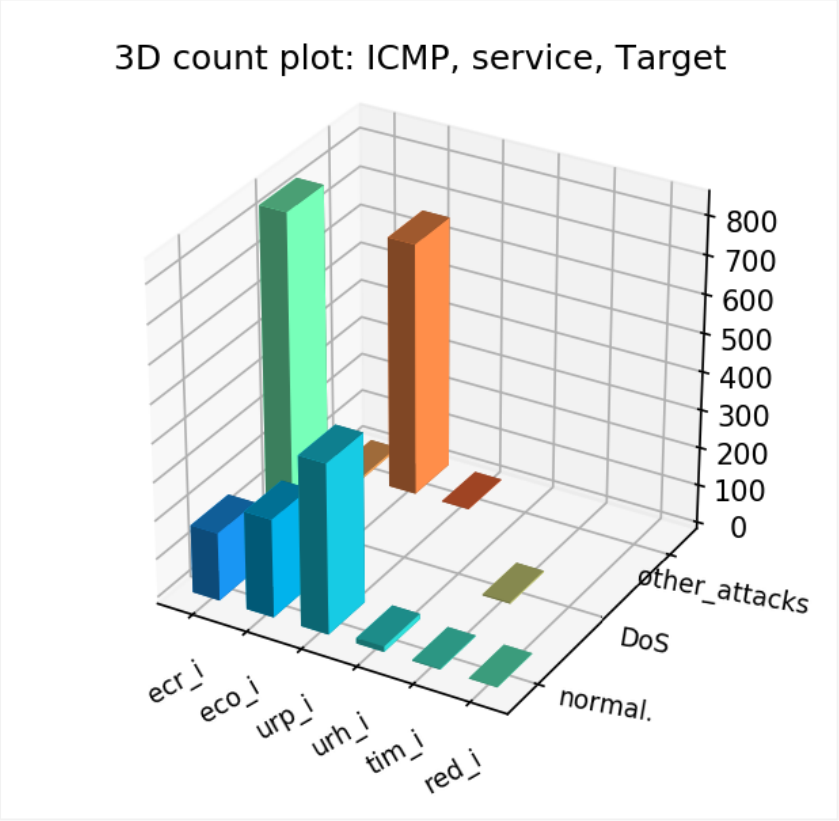


Figure 10: 3D Count plot ICMP, service and Target

Results for Random Forest:

	Correlation-threshold	Detection Rate	Precision	False Negative Rate	AUC	False Alarm Rate
0	0.5	99.83%	99.86%	0.17%	99.87%	0.09%
1	0.6	99.83%	99.90%	0.17%	99.88%	0.06%
2	0.7	99.85%	99.90%	0.15%	99.89%	0.07%
3	0.8	99.87%	99.96%	0.13%	99.92%	0.03%
4	0.9	99.85%	99.94%	0.15%	99.91%	0.04%

Figure 11: Correlation Filter RF

Results for Naïve Bayes WITHOUT Binarization:

	Correlation-threshold	Detection Rate	Precision	False Negative Rate	AUC	False Alarm Rate
0	0.5	93.82%	93.74%	6.18%	94.86%	4.11%
1	0.6	93.82%	93.74%	6.18%	94.86%	4.11%
2	0.7	93.86%	93.69%	6.14%	94.86%	4.15%
3	0.8	93.77%	93.70%	6.23%	94.81%	4.14%
4	0.9	93.88%	99.38%	6.12%	96.75%	0.39%

Figure 12: Correlation Filter BNB without Binarization

Results for Naïve Bayes WITH Binarization:

	Correlation-threshold	Detection Rate	Precision	False Negative Rate	AUC	False Alarm Rate
0	0.5	94.02%	99.73%	5.98%	96.93%	0.17%
1	0.6	94.04%	99.73%	5.96%	96.93%	0.17%
2	0.7	94.66%	99.82%	5.34%	97.27%	0.11%
3	0.8	93.71%	99.88%	6.29%	96.82%	0.07%
4	0.9	93.91%	99.77%	6.09%	96.89%	0.14%

Figure 13: Correlation Filter BNB with Binarization

```
Number of Iterations:  16 , Number of parameter-sets leading to max DR:  1

  Detection Rate Precision False Negative Rate     AUC False Alarm Rate
0         99.89%    99.94%               0.11%  99.92%            0.04%

Computational Time : 31  Minutes
```

Figure 14: Parameter Optimization RF

```
Number of Iterations:  1620 , Number of parameter-sets leading to max DR:  90

The distinct result / results leading to max DR is / are:
  Detection Rate Precision False Negative Rate     AUC False Alarm Rate
0         95.00%    99.77%               5.00%  97.43%            0.15%

Computational Time: 6  Minutes
```

Figure 15: Parameter Optimization BNB

Declaration of Academic Honesty

I, Iliyana Pekova, hereby declare that I have not previously submitted the present work for other examinations. I wrote this work independently. All sources, including sources from the Internet, that I have reproduced in either an naltered or modified form (particularly sources for texts, graphs, tables and images), have been acknowledged by me as such. I understand that violations of these principles will result in proceedings regarding deception or attempted deception.

I, Georg Velev, hereby declare that I have not previously submitted the present work for other examinations. I wrote this work independently. All sources, including sources from the Internet, that I have reproduced in either an naltered or modified form (particularly sources for texts, graphs, tables and images), have been acknowledged by me as such. I understand that violations of these principles will result in proceedings regarding deception or attempted deception.

Berlin, 26.03.2019