PROACTIVE RECOGNITION OF DOMAIN ABUSE

JOOST PRINS

Master of Science (MSc.)
DACS
University of Twente

March 27, 2021

Joost Prins: *Proactive Recognition of Domain Abuse,* Proactive Recognition of Domain Abuse, © March 27, 2021

ABSTRACT

The number of phishing domains increases due to the ever-increasing worldwide internet use. This research contributes to the state-of-the-art of the recognition of phishing domains at the time of registration. This was done in two steps. In the first step, a system that automatically verifies the correctness of the registration information was created. This system aids abuse analysts by giving them information about the correctness of the registration information of a domain. In the second step, a classifier was created using features regarding the registration information, such as the correctness of the information, to detect phishing domains. The results show that such a classifier can detect malicious domains with a performance that is on par with the current state-of-the-art without relying on bulk registration features. The results show that checking the correctness of the registration information of newly registered domains is a useful indicator in predicting whether a domain will be used for phishing purposes. Furthermore, the results can be used to improve the defences and safety of the .nl country code Top Level Domain (ccTLD).

ACKNOWLEDGEMENTS

First, and foremost, I want to thank my supervisors. Especially Moritz, for his continued guidance and feedback throughout the research.

Furthermore, I want to thank my SIDN Labs colleagues for the nice Tuesdays, although few in number due to COVID-19.

Finally, I want to thank my family and friends for their support. Special regards to Kimberly, for her continued support and motivation. My parents, for their support and feedback. And the study friends that listened to me rambling about my research.

CONTENTS

1 BACKGROUND 1
- 1.1 Introduction 1
- 1.2 Motivation and Objectives 2
- 1.3 Contributions 3
- 1.4 Outline 4

2 RELATED WORK 5
- 2.1 Method 5
- 2.2 State-of-the-Art 5
  - 2.2.1 URL Classification 6
  - 2.2.2 Content-based Classification 8
  - 2.2.3 Registration Information Based Classification 10
  - 2.2.4 Rule-based Classification 11

3 METHOD 13
- 3.1 Data Collection 13
- 3.2 Data Validation 15
  - 3.2.1 Name 16
  - 3.2.2 E-mail Address 16
  - 3.2.3 Phone Number 17
  - 3.2.4 Address 17
- 3.3 Data Classification 18
  - 3.3.1 Validation Features 18
  - 3.3.2 Domain Features 18
  - 3.3.3 Pre-Processing Steps 19
  - 3.3.4 Data Set Imbalance 21
  - 3.3.5 Classifier Decision 21
  - 3.3.6 Classifier Evaluation 22

4 RESULTS 23
- 4.1 Data Validation 23
  - 4.1.1 Name 23
  - 4.1.2 E-mail Address 24
  - 4.1.3 Phone Number 25
  - 4.1.4 Address 26
  - 4.1.5 Combination 27
  - 4.1.6 Overall Results 28
- 4.2 Data Classification 29
  - 4.2.1 Validation Features Classifier 29
  - 4.2.2 Extended Features Classifier 30
  - 4.2.3 Random Forest Hyper Parameters 31
  - 4.2.4 Classifier Results 31
  - 4.2.5 Classifier Evaluation 33

5 DISCUSSION 35
- 5.1 Implications 35

5.2 Limitations 36
5.3 Ethical Considerations 36
6 CONCLUSION 38
6.1 Future Work 40
A APPENDIX 41
A.1 Abuse tokens 41
A.2 Erratum 43
BIBLIOGRAPHY 44

LIST OF FIGURES

Figure 4.1 Validation results of the name feature over each data set 24
Figure 4.2 Validation results of the mail features over each data set 25
Figure 4.3 validation results of the phone feature over each data set 26
Figure 4.4 validation results of the address feature over each data set 27
Figure 4.5 Combination results of all the validation features 28
Figure 4.6 Pearson correlation between validation features and the label 30
Figure 4.7 ROC curve and AUC with the random forest classifier 32
Figure 4.8 Feature importances of the random forest classifier 32
Figure 4.9 ROC curve of the random forest on the evaluation data set 33
Figure 4.10 Capital letter count distribution of the test and evaluation data sets 34
Figure 4.11 Address validity distribution of the test and evaluation data sets 34

LIST OF TABLES

Table 3.1 Overview of data sources used in this research 14
Table 3.2 Overview of data sets 15
Table 3.3 Complete feature set 19
Table 3.4 Feature set with the corresponding pre-processing step 20
Table 3.5 Example of the one-hot encoding process 21
Table 4.1 Examples of the name feature with their classification result 23
Table 4.2 Examples of the mail feature with their classification result 24
Table 4.3 Examples of the phone feature with their classification result 26

Table 4.4 Examples of the address feature with their Levenshtein distance 27
Table 4.5 Mean classification metrics of different classifiers 30
Table 4.6 Confusion matrix of the random forest on the evaluation data set 33

ACRONYMS

TLD	Top Level Domain
SIDN	Stichting Internet Domeinregistratie Nederland
ccTLD	country code Top Level Domain
BAG	Basisregistratie Adressen en Gebouwen
BEC	Business Email Compromise
FBI	Federal Bureau of Investigation
NCSC	National Cyber Security Centre
DNS	Domain Name System
SLD	Second Level Domain
SMOTE	Synthetic Minority Over-sampling Technique

1 BACKGROUND

1.1 INTRODUCTION

The economic impact of malicious domains is an ever-growing problem worldwide. Most malicious domains are phishing websites and domains involved in Business Email Compromise (BEC). For example, in the Netherlands, Pathé lost €19 million due to BEC fraud [7]. The Federal Bureau of Investigation (FBI) stated that more than $12 billion was lost worldwide between October 2013 and May 2018 due to BEC [46] [16]. In the Netherlands, the Dutch National Cyber Security Centre (NCSC) stated that phishing is one of the most prevalent causes of financial losses [29].

There are several ways that perpetrators execute their phishing attacks. Mostly they use text messages or email to contact their victims. However, to make the phishing attack convincing, the perpetrators almost always use a website to actually perform the phishing attack. For a website to function properly, a domain is needed. Perpetrators acquire such domains either by hacking an existing domain or by registering a new domain, specifically for malicious purposes.

There are many ways to combat phishing attacks, such as, by analyzing the contents of a website or analyzing the Domain Name System (DNS) traffic of that domain. All these techniques require the domain to be online and functioning already. Before the domain is detected using these techniques and taken offline, damage could be done in the meantime. Victims may be phished before the website is taken offline. Therefore, this research focusses on the possibility to predict whether domains become malicious at the time of registration. If such a system performs well, phishing websites could be taken offline before they could do any damage. This was done by implementing a line of defence at a ccTLD. For this research, the .nl ccTLD was used, because Stichting Internet Domeinregistratie Nederland (SIDN) was the collaborating company for this research.

The process of registering a second-level domain at a Top Level Domain (TLD) involves three parties: the registrant (the person/company that wants to buy a second-level domain), the registrar (the company that sells such second-level domains to the registrant), and the registry (the company that manages the TLD, SIDN for the .nl ccTLD). When a registrant purchases a second-level domain from the registrar, the registrar further handles the request with the registry. When the registry approves the request for the domain, the domain will be allocated, the database of the TLD is updated, and the appropriate DNS records will be set at the DNS of the TLD.

The implementation of the line of defence was a two-step process. First, a system was created to automatically detect false registration information. Second, a system was developed that uses the registration information, such as the correctness thereof, to detect phishing domains. Information that is available at the time of registration of .nl domains, such as the name, phone number, and email address of the registrant was used for this purpose.

1.2 MOTIVATION AND OBJECTIVES

In the Netherlands, SIDN is the organisation that issues .nl domains and maintains a register of the owners of all .nl domains. To register a domain, the registrant needs to provide personal details to obtain the domain.

SIDN has two ways to fight the abuse of a domain. The first way is by terminating the domain on the grounds of the incorrectness of the registration information (article 16) [9]. The second way is by terminating the domain on the grounds of abusive behaviour (article 18) [9]. It is more difficult to prove that a domain has abusive behaviour than that it has incorrect registration information. Due to this, the abuse team of SIDN also uses article 16 to fight abuse.

The assumption that is made in this research is that malicious domains are not registered with real registration information to avoid the chance of being traced. Currently, SIDN has no automated system that checks the validity of the registration information. The validity of the registration information of a domain is checked when suspicious activity is found involving a particular domain. When it is suspected that the registration information of a domain is incorrect, SIDN asks the domain name holder and the registrar for proof of correctness of the registration information [44]. If the correctness of registration information is not provided within 5 days, the domain can be taken offline.

From the literature review, discussed in Section 2.2, several conclusions can be drawn. First, all the discussed methods try to detect malicious intent at different stages of the domain life cycle using different aspects of the domain. Furthermore, all the different methods do not perform perfectly. However, using these methods together improves the protection against domains with malicious intent. Second, little research has been done into using information available at time of registration to detect domains with malicious intent. Third, for the systems that do try to predict malicious domains at time of registration, such as the PREMADOMA system [41], the correctness of the registration information is not discussed. This research hypothesizes that the correctness of the registration information is a discriminating indicator of domains with malicious intent. Finally, both PREMADOMA [41] and PREDATOR [12] focus on detecting large-scale campaigns, such as the phishing campaign targeting Canadian banks where over 300 different domains were used [22], by using features regarding bulk registration characteristics as predictors for malicious intent. Some examples of bulk registration characteristics are the reuse of information such as the e-mail address or the phone number. Therefore, in this research, the scope targets the gap in the current state-of-the-art, namely small scale campaigns that only utilise one domain, and therefore are not part of bulk registrations.

The research has two objectives. The first objective is to design a system that automatically classifies the registration information as correct or not. The second objective is to design a system that classifies new registrations as either a phishing domain or benign domain using features regarding the correctness of the registration information.

The following research questions have been formulated:

RQ1 How can false registration information detection be automated?

- RQ1.1 Which parts of the registration information can be validated and are a useful discriminator to detect false registration data?
- RQ1.2 How can the validation of registration information from ***RQ1.1*** be automated?

RQ2 Can small scale phishing domains be detected at time of registration for second-level domains in the .nl top-level domain?

- RQ2.1 Is false registration information a good predictor of the future use of the domain for phishing?
- RQ2.2 What other features are good predictors of the future use of a domain for phishing?
- RQ2.3 What is the detection performance of phishing domains in the .nl top-level domain of a classifier based on the features identified in ***RQ2.1*** and ***RQ2.2***?
- RQ2.4 What are the best model parameters such that the classifier of ***RQ2.3*** performs the best according to SIDN?

1.3 CONTRIBUTIONS

This research has three main contributions. First, a classifier that can automatically detect false registration information. This classifier supports the research done by abuse analysts. On average, 6000 to 10000 new domains are registered each day at the .nl ccTLD. With the classifier, these analysts do not have to crawl through the bulk of registration information manually to find a needle in a haystack, but instead, the classifier can provide analysts with a shorter list of domain names which do not have correct registration information, acting as a starting point for the detection of phishing domains. Furthermore, the output of this classifier can be used as a new feature for the detection of domains with malicious intent. Second, the advancement in knowledge of mitigation of domains with malicious intent. PREMADOMA mainly focuses on the detection of large-scale campaigns. Which means that they miss domains that are not registered in bulk, such as phishing websites. Therefore, in this research, the focus lies in the detection of such domains that are used for phishing. Third, it yields a classifier that can detect domains used for phishing purposes, trained specifically for the .nl ccTLD. This classifier helps improve the defences and make the .nl ccTLD safer and more trustworthy.

1.4 OUTLINE

Chapter 2 gives an overview of existing research into the prediction of malicious domains based on the registration information.

Chapter 3 provides an overview of the methodology. This chapter is split into two parts, the first part discusses the validation process, the second part discusses the classification process.

Chapter 4 presents the results of the research in two parts. The first part presents the results of the validation process. The second part presents the results of the classification process.

Chapter 5 discusses the implications and the limitations of this research

Chapter 6 concludes the research, answers the research questions, and presents opportunities for future work.

2 RELATED WORK

This chapter describes the current state-of-the-art regarding malicious domain detection. It is split into two parts; Section 2.1 describes the method used for the analysis of the current state-of-art and Section 2.2 presents the results of the analysis of the current state-of-art.

2.1 METHOD

To perform a successful and complete literature study on the topic of malicious domain detection, a methodology based on the Structured Literature Review of Kitchenman [20] is used. First, a complete search query is defined to find relevant articles. The search query is defined as:

```
("detection" OR "classification" OR "recognition") AND ("URL" OR
    "registration information") AND "malicious" AND NOT (("image"
     OR "picture") OR "cell" OR "receptor" OR "genetics" OR "
    botnet" OR "DDoS" OR "scripting" OR "social" OR "twitter")
```

From the search query, several semantic changes were made such that the query functions as expected in combination with different databases. The following databases were used to search for relevant literature:

- Scopus (212 results)
- Scholar (110 results)
- IEEE (86 results)
- Web of Science (99 results)

Furthermore, when analyzing the resulting literature, the references of the literature were used to find additional literature.

The search resulted in 507 papers. After reading the title, abstract, and conclusion, 49 were deemed relevant based on the language, subject, and conclusions of the paper. After a full review, 23 papers were used in this literature review based on the approach and results of the papers.

2.2 STATE-OF-THE-ART

The research area of malicious website detection has been studied extensively. There is a wide variety of strategies applied to detect malicious websites. This section provides an overview of the different strategies and is divided into four parts. The first part describes the work that has been done into the detection of malicious pages only using the Uniform Resource Locator (URL) of the page. The second part provides an overview of the work done into analyzing the actual content of a website to determine whether the page is malicious or not. The third part discusses research into using the registration data of a domain to determine its validity. The fourth part dives into work that has been done using rule-based classification approaches (e.g. blacklists) to detect malicious pages.

2.2.1 *URL Classification*

The URL plays a key role in phishing activities, as it is the link between the end-user and the phishing website. The research area of using the URL to detect malicious intent of a website has been studied by [17, 19, 21, 23, 32, 35, 38, 47–49, 51]. Many different machine learning algorithms have been applied to the problem of classifying URLs as either malicious or benign. Furthermore, in recent years, another dimension was added to this problem with the rise in URL shortening services [19].

It is important to have a large and balanced data set to establish a well-performing URL classifier [34]. Several different data sources are used in the current state-of-the-art. Benign URLs are retrieved from search engines, such as the Yahoo most visited sites [17, 24]. Malicious URLs are gathered from blacklists, such as PhishTank [17, 24], and SpamScatter [24].

To achieve high accuracy on the classification of malicious URLs, the right features need to be used. Jeeva et al. used rule mining to determine the most important features of phishing URLs [17]. Two different rule generating algorithms were used: a priori and predictive a priori. The rule generating algorithms revealed that features regarding transport layer security, the availability of the TLD, and certain keywords in the URL were most discriminating. In another study, Li et al. used linear and non-linear space transformation of features to improve the accuracy of traditional classifiers [23]. They used four different categories of features: (1) domain-based features, (2) host-based features, (3) reputation-based features, and (4) lexical features. They identified three problems regarding the features for traditional machine learning classifiers: (1) there is a strong correlation between certain features, (2) the weight/importance of each feature is unknown, and (3) the size of the data set is generally too large to apply kernel methods for linear classifiers. From these findings, they propose a method to transform features in the pre-processing step for the URL classification problem. The method consists of three phases: (1) grid search for optimal parameters, (2) single value decomposition, and (3) space transformation. The results show that this method improves the performance of a wide variety of machine learning algorithms.

Several studies also looked at the possibility of using algorithms to automatically extract and select the most relevant features. Several different algorithms are used for feature extraction in the current state-of-the-art. Some examples are a stacked restricted Boltzmann machine [38], a cuckoo search algorithm [35], a multicore convolutional neural network [48], and a traditional convolutional neural network [32].

Several studies have applied traditional machine learning algorithms to the challenge of detecting malicious URLs. Ma et al. use lexical (e.g. length of the URL) and host-based (e.g. IP address and WHOIS properties) features for the classification of malicious URLs [24]. Three different machine learning algorithms were tested: (1) Naïve Bayes, (2) Support Vector Machines, (3) Logistic Regression. It followed that the Logistic Regression classifier performed the best with an FP rate of 7.6% and a TP rate of 0.1% regarding the classification of malicious URLs. In a follow-up study, Xiong et al [47]. used a tri-gram detection algorithm. This algorithm was faster and had a lower FP rate than the Logistic regression classifier of Ma et al [24].

In other studies, ensemble methods were used to tackle the problem of URL classification. Khan et al. created an AdaBoost algorithm that classifies an URL into different categories, such as trojan, mebroot, and exploits [19]. Furthermore, Kumar et al. constructed a layered detection model [21]. This model consists of four different types of classifiers: (1) black/white list filter, (2) Naïve Bayes filter, (3) CART decision tree filter, and (4) Support Vector Machine filter. Using this layered detection model, they were able to achieve an accuracy of 79.55%.

In a study by Zhao et al., they argue that using traditional machine learning techniques cannot be used for the classification of URLs as the importance of time is not captured with a traditional machine learning algorithm [51]. Therefore, they propose a cost-sensitive online active learning approach. The approach reaches similar performances as state-of-the-art techniques, but with a much lower computational cost.

Due to the advancements in processing power in recent years, deep learning has become a viable option in many different fields as a classification algorithm. Several studies have applied the use of deep learning to the classification of URLs. Selvaganapathy et al. use a deep learning network that achieves an accuracy of 75% [38]. Peng et al. use a Long-Short Term Memory network that achieves an accuracy of 98% [32]. And Yang et al. created a Convolutional Gated-Recurrent-Unit Neural Network which achieved an accuracy of 99.6% [48].

The use of only URLs to classify domains as either malicious or benign shows excellent results. Thus, it can be stated that using URLs as a detection mechanism for malicious domains is an effective strategy.

However, this research aims to detect domains as malicious at time of registration. While the domain name is part of the URL and is recorded at the time of registration, the aforementioned URL classification strategies utilize additional parts of the URL as well, such as the path component. These additional parts only become available when the domain is hosting a website, which is later than the time of registration. Therefore, the mentioned methods can not fully be utilized in this research. However, the overall goal of URL classification is in line with the current research, namely, detection of abusive domains. For this reason, two aspects of the reviewed research are used in this research. First, abuse feeds are used as ground truth for malicious domains in the data collection part of the research. Second, several machine learning classifiers were also tested in the research to test their effectiveness.

2.2.2 *Content-based Classification*

Another research area that has been studied extensively is the use of page contents to identify malicious websites. CANTINA was the pioneer in this research area [50]. CANTINA applies the Term Frequency / Inverse Document Frequency (TF-IDF) algorithm to the problem of identifying phishing websites. CANTINA identifies the most frequently used terms on the website. From these terms, a lexical signature is generated. The lexical signature is then fed into a search engine and checks whether the website is in the top result of the search. CANTINA achieved a 97% true positive rate with a 6% false-positive rate.

To create a classifier that is effective in identifying malicious websites, different features need to be considered carefully. Singh et al. research 25 different features and conclude 5 of those to be the most discriminating: (1) use of cloaking, (2) presence of an IFrame, (3) presence of redirection rules, (4) size of obfuscated code, and (5) amount of Popups using the Window.open() function. These features were tested on two different machine learning classifiers: C4.5 and Naïve Bayes. In another study, Kazemian et al. test a wider variety of features with different characteristics [18]. The experiments show that features regarding the content of the page are the most discriminating, after which the URL and visual features (such as images) were the most important. In addition to identifying the most discriminating characteristics, Kazemian et al. tested three supervised and two unsupervised machine learning techniques. Their experiments show that the supervised learning techniques reach over 89% accuracy and the unsupervised learning techniques reach a silhouette coefficient of 0.87 with the most discriminating feature set.

Following CANTINA, several different systems were developed that tested traditional machine learning techniques to identify malicious websites based on their contents. Manek et al. created DeMalFier [25]. DeMalFier uses features regarding the content of the page, URL information, and host information as features. Three different machine learning models were tested: Naïve Bayes, Logistic Regression, and Support Vector Machines. The experiments showed that Logistic Regression was the best performing model with an accuracy of 99%.

Different studies with different features showed that the Support Vector Machine was the best performing machine learning classifier. Chiba et al. achieve an accuracy of 90% with only using features regarding the IP address [8]. Bannur et al. achieve a precision of 97.6% and recall of 96.6% with URL, the structure of the web page, and visual features [4]. Canali et al. report a false positive rate of 5.46 and a false negative rate of 4.13 with their system ProPhiler [6]. ProPhiler uses features regarding the HTML content of the page, the JavaScript on the page, and the URL.

The experiments of Messabi et al. showed that the C4.5 algorithm worked best for their feature set [2]. The feature set consisted of features regarding the domain, character indicator variables, and tokens. The C4.5 algorithm was able to achieve an accuracy of 77.5%.

Rao et al. reported that the Random Forest classifier performed the best out of 8 different machine learning classifiers [36]. The feature set of Rao et al. consists of features regarding the URL, third-party variables, and hyperlinks. The Random Forest classifier was able to achieve an accuracy of 99.31%.

In other studies, other classification methods were explored. Using Boyer-Moore string pattern matching, Gupta et al. were able to achieve an accuracy of 85% [11]. Aburrous et al. explored the possibility of using fuzzy techniques to detect phishing websites [1]. They tested 26 different features regarding the source code of the page, information surrounding the page, and the certificate of the website. Their experiments show that a fuzzy technique based approach could work in identifying phishing websites.

Apart from static analysis techniques where machine learning classifiers are used, the use of dynamic analysis techniques is researched. Moshchuck et al. developed SpyProxy which is a system that combines a static analysis check and a Virtual Machine based check [26]. The Virtual Machine based check tries to identify suspicious activity outside of the browser sandbox. If such behaviour is seen outside the browser sandbox, then the site is unsafe. If such behaviour is not seen, then the site is safe. From the tests, it showed that SpyProxy was able to successfully identify every malicious website out of the 100 tested websites.

Besides the traditional machine learning techniques, Torroledo et al. applied a deep learning network to identify malicious websites [42]. They use over 30 different features based on the TLS certificate of a website that feed the deep learning network. The deep learning network is a recurrent neural network with a long short-term memory layer. The deep learning network was able to achieve an accuracy of 94.87%.

The research shows that content-based classification is an effective tool to combat domains with malicious intent when the domain is presented to the classifier. However, this research area has the same downside identified in Section 2.2.1, where domains with malicious intent can only be identified when a domain already hosts a website and not at time of registration.

From the research, several things were learned. First, domain features are a useful indicator of whether the domain would become malicious or not. Second, a better overview of which machine learning classifiers could be used to classify malicious domains was gained.

2.2.3 *Registration Information Based Classification*

There are two studies using data available at the time of registration of a domain to classify its intent, namely: PREDATOR [12] and PREMADOMA [41].

PREDATOR explored the possibility of using data available at the time of registration to identify malicious domains [12]. PREDATOR was designed to be used by registries and registrars. Three key insights were made during the case study: (1) domain registrations often occur in bursts, (2) domains registered together are often at a similar stage in the domain life-cycle, for example, whether the registration is a brand new registration, or a re-registration after some time that the domain was expired, and (3) domains registered together may be similar to one another, for example, when looking at the brand new malicious registrations, domain names appear similar to one another, due to having the same substring across different domain names. From the case study, meaningful features were selected. These features can be divided into three categories: (1) domain profile, (2) registration history, and (3) batch correlation. The data for these features are all found in the zone files. PREDATOR uses a Convex Polytope Machine to classify domains as either malicious or benign. Using this setup, PREDATOR can achieve a detection rate of 70% with a false positive rate of 0.35%.

Following PREDATOR, PREMADOMA built upon the idea of using data available at time of registration [41]. The difference between PREDATOR and PREMADOMA is their different feature set and the use of different classification algorithms. Furthermore, PREMADOMA uses features regarding the registrant data of a domain, whereas PREDATOR does not. Registrant data consists of, for example, the name, email address, and phone number of the registrant. The classifier that PREMADOMA uses is a combination of a PART algorithm that classifies based on features regarding the reputation of the registrant data and an agglomerative clustering algorithm that classifies based on similarity features. The reputation of features is determined by the percentage of malicious registrations that are linked to that feature in the ground truth data. The similarity of a domain to a malicious cluster is calculated using the pair-wise distance for each feature in the data. For strings, the normalized Levenshtein distance is used, for numerical values, the euclidian distance is used, and for categorical features, the similarity distance is expressed as either 0 or 1. From the experiments, PREMADOMA was able to achieve a recall of 66.23% with a precision of 84.57% and a false positive rate of 0.3%. After the experiments, PREMADOMA was deployed in the .eu ccTLD as the first line of defence in the battle against malicious domains. During the 17 months deployment of PREMADOMA in the .eu ccTLD, it was able to take down 58,966 malicious domains.

The PREMADOMA system has good performance in the .eu ccTLD. To limit the number of false positives when automatically assessing domain registrations, the decision was made to cap the false positive rate to 0.3%.

PREMADOMA [41] and PREDATOR [12] show that it is possible to predict whether domains will become malicious at the time of registration of a domain. From the research, several things were used in this research. First, the use of fuzzy matching in several features in the feature set to negate typos, for example, the given address. Second, ideas for features were gained.

2.2.4 *Rule-based Classification*

Rule-based classification is the practice of using existing rules to identify URLs as malicious or benign. Currently, there are two ways that this is done: blacklisting and whitelisting. Both blacklists and whitelists can be used to filter URLs. Although using blacklists and whitelists is not a detection method, they contribute to the safer use of the internet. A blacklist contains domains that are malicious and should not be visited. A whitelist contains domains that are safe to visit. Due to the large volume of domains available, whitelisting techniques are not feasible whereas blacklists can be deployed in multiple settings, namely: desktop applications, browser extensions, and router implementations. Multiple providers of such blacklists exist, such as Google Safe Browsing [37] and Microsoft Smart Screen [39]. A major disadvantage of blacklisting services is the fact that domains only occur on the blacklist after it has already been used in malicious activities [24].

Nandhini et al. explore the possibility of implementing bloom filters as a client-side blacklisting application [28]. Bloom filters have the benefit that the size of the data set does not heavily impact the speed of the classification. Therefore, it is beneficial to use bloom filters when dealing with large blacklists. Their experiments show that such an implementation only adds 0.1 seconds of loading time per page, and therefore does not impede normal browsing of the internet.

Black- and whitelists are a good countermeasure to contain detected malicious domains. However, this can only be done when the malicious domain has already caused damage to users. Additionally, most phishing domains are part of a hit-and-run strategy where the lifespan of a domain is extremely short [13]. Because of the hit-and-run strategy, black- and whitelists have become a less effective countermeasure against domains with malicious intent. Furthermore, because the hit-and-run strategy has become a more widely used method by perpetrators, it has become more important to detect and take down malicious domains as early as possible.

3 METHOD

This chapter lays out the method of the conducted research. It is divided into three parts. Section 3.1 describes the method taken to collect and process the data used for the research. Section 3.2 describes the method used to answer research question 1. Section 3.3 describes the method taken to answer research question 2. This research has been approved by the ethics committee of the University of Twente[1] and the privacy board of SIDN.

3.1 DATA COLLECTION

The main source of data for this research project is gathered from the registration database (DRS). This database contains all the active and historic .nl domains with their registration information. There is no delay from when a domain is registered, to when this registration is available in DRS. Registration data that is available about a domain from DRS is:

- Domain name
- Time of registration of the domain
- End date of the registration of the domain
- Registrar with which the domain was bought
- Full name of the person/company registering the domain
- The email address of the person/company
- The administrative email address that would need to be used in case SIDN wants to contact the person responsible for the domain
- the phone number of the person/company

For this research, three more data sources are used. First, a central feed of abuse feeds. This feed contains the aggregation of the following three abuse feeds: (1) APWG [3], (2) URLhaus [43], and (3) Phishtank [33]. Second, data from the abuse team of SIDN. Over the years, the abuse team has documented their actions against 4769 different domains with malicious intent. The data from the abuse team and abuse feeds are used as ground truth for the research. The third, and final,

[1] reference number RP 2020-68

source of data is the Basisregistratie Adressen en Gebouwen (BAG) register. BAG is the open-source register of the Dutch government where all address information is stored of every address in the Netherlands. The BAG is used to verify whether the address information that is provided, is correct. Table 3.1 gives an overview of all the different data sources and their purpose.

Table 3.1: Overview of data sources used in this research

Data source	Description	Usage
DRS	Registration database	Collect relevant information about specific domain names
Aggregated abuse feeds	List of domain names that were used for abuse purposes	Use as ground truth data for the classification part of the research
Abuse team	List of domain names that the abuse team of SIDN has acted against over the years	Use as ground truth data for the classification part of the research
BAG	Database of all addresses in the Netherlands	Use to check the validity of addresses of Dutch domain registrations

The following data sets are used for the validation of the different registration features, and the classification of the domains. An overview of the data sets can be found in Table 3.2:

ABUSE TEAM data set contains all the domains that the abuse team of SIDN have flagged during a period of 5 years. The goal of the abuse team is to fight domains that have malicious intent, not domains with incorrect registration features. They only look at the registration information when they suspect that the domain is used for malicious purposes based on the website. If the abuse team concludes that the domain is used for malicious intent, they use the false registration information as a tool to take the domain offline. Because of this, most of the domains within the abuse team have false registration features. The data set contains around 4600 unique domains with around 2900 unique registration features.

ABUSE FEED data set contains all the domains that have been flagged by the different abuse feeds that SIDN gathers. The data set contains around 1900 unique domains with around 1600 unique registration features.

RANDOM data set contains a randomly selected sample set of currently active domains in the .nl ccTLD that have been active for over 30 days. According to Vissers et al. [45], 98,57% of abusive domains are on a blacklist after 30 days. For this reason, the constraint of having been active for more than 30 days has been chosen to reduce the number of malicious domains in this data set. The data set contains around 91000 unique domains.

TRUSTMARK data set contains domains that have been vetted as being legitimate webshops and have a quality mark by Trustmark [14]. This data set provides vetted domains, which are used for legitimate purposes. The data set contains around 71000 unique domains.

Table 3.2: Overview of data sets

Data set	Usage	Number of domains
Abuse team	Malicious	4600
Abuse feed	Malicious	1900
Random	Benign	91000
Trustmark	Benign	71000

3.2 DATA VALIDATION

The first step in answering the first research question *how can false registration information detection be automated?*, is answering the first subquestion *which parts of the registration information can be validated and are a useful discriminator to detect false registration data?*. To answer the first subquestion, each registration information feature that is available in the registration database is examined and the possibility of validating the information is evaluated.

When each feature in the registration database is evaluated and the possibility of validating the feature is determined, the possibility of automating the process of the validation of these features is determined.

The following registrant data is available when a new domain is registered in the .nl ccTLD:

- Full name of the person/company registering the domain
- The email address of the person/company
- The administrative email address that would need to be used in case SIDN wants to contact the person

- the phone number of the person/company

The assumption that we make is that the more incorrect details are provided by the registrant, the more suspicious the domain registration is. For example, registrants can make a typo, but when all their details are incorrect, this is more suspicious.

For each information feature, the method of validating this information and the automation of the validation is evaluated and discussed in the following sections.

3.2.1 *Name*

The name feature contains the full name of the registrant. This can be a company name or natural person. The registration data contains another feature that says whether the registration information is for a person or a company.

The validation of the name feature is done using SpaCy [40]. SpaCy uses a deep learning model to tokenize natural language and recognize named entities. Given the fact that any nationality can register a .nl domain, the deep learning model of SpaCy that is used was trained on a 2010 corpus containing Wikipedia [27]. This was done to ensure multi-language support for the recognition of the name feature.

After initial testing with the SpaCy library using dutch company names and personal names, it was concluded that SpaCy can detect Dutch personal names with a high accuracy but not company names. Therefore, only natural persons are validated.

If SpaCy recognizes a natural person entity in the full name feature, the name is valid. Otherwise, the name feature is not valid.

3.2.2 *E-mail Address*

For each domain that is registered, three different mail address fields are present. The mail address of the domain holder, the mail address of the administrative contact, and the mail address of the technical contact. The mail address can be validated by establishing a connection to the mail server of the mail address, and if there is no error with setting up the connection, the mail address is correct.

The mail address is validated by checking whether the host of the mail has a valid SMTP server by sending an SMTP HELO message. Furthermore, to check whether the mail address exists, the RCPT TO is set to the mail address. If the mail server returns a 250 OK, the mail address exists.

For the validation of the mail address, an SPF record was set up with a test domain to make sure the receiving servers did not deny the request based on having incorrect SPF records set.

3.2.3 *Phone Number*

There are two types of phone number validation methods. The first uses online services to validate a phone number, like numverify [30]. The second approach only checks whether the format of the phone number is correct and possible. This can be done by using a local library, for example, the Google phonenumbers library [10].

Due to the sensitive nature of the registration information, online services are not used to verify the phone numbers of registrations.

3.2.4 *Address*

The following information is available of the address:

- Full address (street, street number, and any suffixes)
- Postal code
- City
- Country

The validity of the address could be checked by querying parts of the information into a search engine (e.g. Google) and check whether the additional information, such as the postcode, city, and country, matches the given information. Another option is to use a dedicated map service, like OpenStreetMap [31], for the same purpose. Additionally, for Dutch addresses, using the BAG database is a possibility. As described in Section 3.1, SIDN has a copy of the BAG database in their server cluster available for this use.

After testing the usage of OpenStreetMap for the validation of international addresses, it was concluded that the usage of OpenStreetMap gave false results. Due to this and the fact that around 95% of the address information is for Dutch addresses, only Dutch addresses are validated with the use of BAG.

When an address is validated, the street number and suffix is split from the street. After the split, the street number along with the postal code is used to query the BAG database. The result from the BAG database is concatenated into a single string containing: full address, postal code, city, and country. This string is then compared to the original address. The comparison of the strings is done using the Levenshtein distance between each substring. The Levenshtein distance expresses the number of changes needed to transform one string into another string with as minimum changes as possible. The Levenshtein distance is used to negate typos that were made when entering the street name or city.

3.3 DATA CLASSIFICATION

This section describes the methodology used to answer the second main research question *can small scale phishing domains be detected at time of registration for second-level domains in the .nl top-level domain?* The first step in answering this research question is answering the first and second sub-questions *is false registration information a good predictor of the future use of the domain for phishing?* and *what other features are good predictors of the future use of a domain for phishing?*. The method of the first sub-question is described in Section 3.3.2. The method of the second sub-question is described in Section 3.3.2. Using the results of Section 3.3.1 and Section 3.3.2, the third sub-question *what is the detection performance of phishing domains in the .nl top-level domain of a classifier based on the features identified in Section 3.3.1 and Section 3.3.2?*, and fourth sub-question *What are the best model parameters such that the classifier of **RQ2.3** performs the best according to SIDN?* can be answered.

3.3.1 *Validation Features*

From Section 3.2, four different validation features of the registration information were identified and created. To test the importance of these features in a classifier two steps were taken. The first step is calculating the correlation of each feature to the label. This expresses the linearity of each feature to the label. A higher correlation means that the feature can say more about the label than a lower correlation. The second step is testing and training a random forest classifier to research the performance of a classifier based on only these features.

3.3.2 *Domain Features*

Apart from the validity features described in Section 3.3.1, features regarding the domain itself are used to improve classification performance. The additional features that are defined are based on features that were used in previous research, namely PREMADOMA [41] and PREDATOR [12]. Additionally, after multiple meetings with the abuse team of SIDN, their recommendations were also used in the selection of the feature set.

Furthermore, an abuse word count feature is used. This feature describes how many words in the domain name are present on a predefined abuse word list. The abuse word list was generated by taking the most occurring words of the domains in the abuse data sets, that were not as occurring in the normal data sets. This was done by using an adjusted version of the Tf-Idf algorithm to generate a score for all the words occurring in the abuse data sets. The function that was used to determine a weight for a word is described in Equation 3.1 where Score is the eventual word weight, x is the word that is checked,

a_x is the total occurrences of that word in the abuse data sets, and o_x is the total occurrences of that word in the normal data sets.

$$Score = \log(\frac{x/a_x}{x/r_x}) \tag{3.1}$$

After the generation of each word weight for each word in the abuse data sets, the 300 highest weighing words were taken, and a selection by hand was made to ensure only relevant words were used. The resulting word list contains 68 words. The exact word list can be found in Section A.1.

An overview of the final feature set is given in Table 3.3.

Table 3.3: Complete feature set

	Feature	Type	Description
Domain	Number of digits	Continuous	The number of digits in the Second Level Domain (SLD)
	Domain length	Continuous	The total length of the SLD
	Contains dash	Boolean	Whether a "-" is in the SLD
	Abuse token count	Continuous	The number of different words that are within the domain name that are in the abuse token word list
Registration	Hour	Discrete	The hour of the day in which the domain was registered
	Weekday	Discrete	The day of the week in which the domain was registered (from 0 to 6)
	Registrar	Nominal	The registrar with which the domain was registered
Name	Word count	Continuous	The number of words that are within the name of the registration information
	Capital letter count	Continuous	The number of capital letters that are within the name of the registration information
Combined	Mail validity	Nominal	Whether the supplied email address is an existing mail address
	Address validity	Nominal	Whether the supplied address is an existing address
	Phone validity	Nominal	Whether the supplied phone number has a correct syntax

3.3.3 *Pre-Processing Steps*

Using the feature set that is composed in Table 3.3, these features first need to be pre-processed for the use with specific classifiers. The pre-processing steps for each feature are described in Table 3.4.

Table 3.4: Feature set with the corresponding pre-processing step

	Feature	Pre-processing step
Domain	Number of digits	Standardize
	Domain length	Standardize
	Contains dash	One-hot encode
	Abuse token count	Standardize
Registration	Hour	Standardize
	Weekday	Standardize
	Registrar	One-hot encode
Name	Word count	Standardize
	Capital letter count	Standardize
Validation	Mail	One-hot encode
	Address	One-hot encode
	Phone	One-hot encode
	Name	One-hot encode

3.3.3.1 *Standard Scaler*

Features need to be standardized to make sure certain machine learning algorithms can handle the information (such as linear models that use L1 and L2 regularization). Standardization of the feature is done by calculating the mean (μ) and standard deviation (σ) of the data. Then for each data point, Equation 3.2 is calculated to standardize the data point, where z is the new value, and x is the original value.

$$z = \frac{x - \mu}{\sigma} \tag{3.2}$$

3.3.3.2 *One-hot Encoding*

For many different machine learning algorithms, input features need to be numeric. Furthermore, the categorical features in the feature set do not have a natural order in the data. Because of this, these categorical features are one-hot encoded. The one-hot encoding process is visualized in Table 3.5. From Table 3.5, the first data entry (index 0) with original value False, is one-hot encoded to the new feature (mail_validity false) being 1, and the others (mail_validity true and mail_validity unknown) being 0. With the one-hot encoding step, the dimensions of the data set change. Because there are a great number of different registrars in the .nl ccTLD, the dimensions of the data set drastically expand from 14 to 919.

Table 3.5: Example of the one-hot encoding process

index	mail_validity
0	False
1	True
2	unknown

⇓

index	mail_validity false	mail_validity true	mail_validity unknown
0	1	0	0
1	0	1	0
2	0	0	1

3.3.4 *Data Set Imbalance*

Because there is an imbalance in the data set that is used (as described in Section 3.1), it is important to balance the data set before training the classifiers. Otherwise, the classifiers gain a bias for the majority class (benign domains). For this research, it was chosen to use Synthetic Minority Over-sampling Technique (SMOTE) to balance the data set [5]. SMOTE works by synthetically generating new samples in the minority class. The generation is done by randomly selecting a data point x, then randomly selecting one of x's nearest neighbours b. When x and b are selected, a line is drawn between these two data points and a random point on this line is chosen as a new data point.

3.3.5 *Classifier Decision*

Different machine learning classifiers have been tested to see which performed best. The selection of machine learning classifiers was made based on previous work described in Section 2.2. The machine learning classifiers that have been tested are:

- Random Forest
- Logistic Regression
- Naïve Bayes
- K-nearest neighbours
- Decision Tree
- Neural networks

3.3.6 *Classifier Evaluation*

Finally, apart from the classifier testing by splitting the data set into a train and test part. The final classifier was tested on new domain registrations. The new registrations were collected in the week of 02/08 until 09/08. After a month, the collected domains that were on any of the abuse feeds were flagged as malicious, others were flagged as benign domains. The data set contains 50701 domains of which 48 have occurred on an abuse feed. Using the combination of data sets from Table 3.2, the final random forest classifier is trained. This classifier is then tested on the newly collected registration data set.

4 RESULTS

This chapter presents the results of the conducted research. It is divided into two parts: Section 4.1 presents the results of the first main research question and its sub-questions. Section 4.2 presents the results of the second main research question and its sub-questions.

4.1 DATA VALIDATION

This section presents the results of the data validation. Each registration feature that was able to be validated is discussed individually in the following sections. For each feature, the distribution of the validity over the unique domains in each data set has been plotted. Namely, the abuse team data set, the abuse feed data set, the random data set, and the trustmark data set. For malicious data sets, an [x] is behind the name and for benign data sets, an [✓] is behind the name.

4.1.1 *Name*

Figure 4.1 shows the validity (as described in Section 3.2.1) distribution of the name feature for each data set. The figure shows that the abuse team and abuse feed data sets have a higher validity than the trustmark and random data sets. Some examples of name features with their validity are given in Table 4.1.

Table 4.1: Examples of the name feature with their classification result

Name	Validity
Sander Rietmeyer	True
liu xuemei	True
Sebastiaan Korse	True
MINNANO-DOMAIN REGISTER SERVICE	False
HomeSecurityXL	False

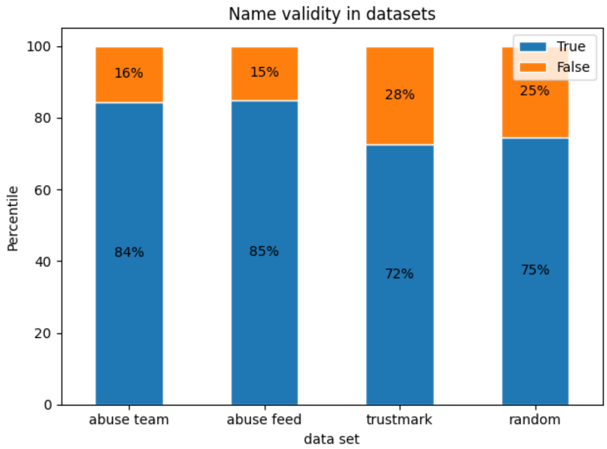


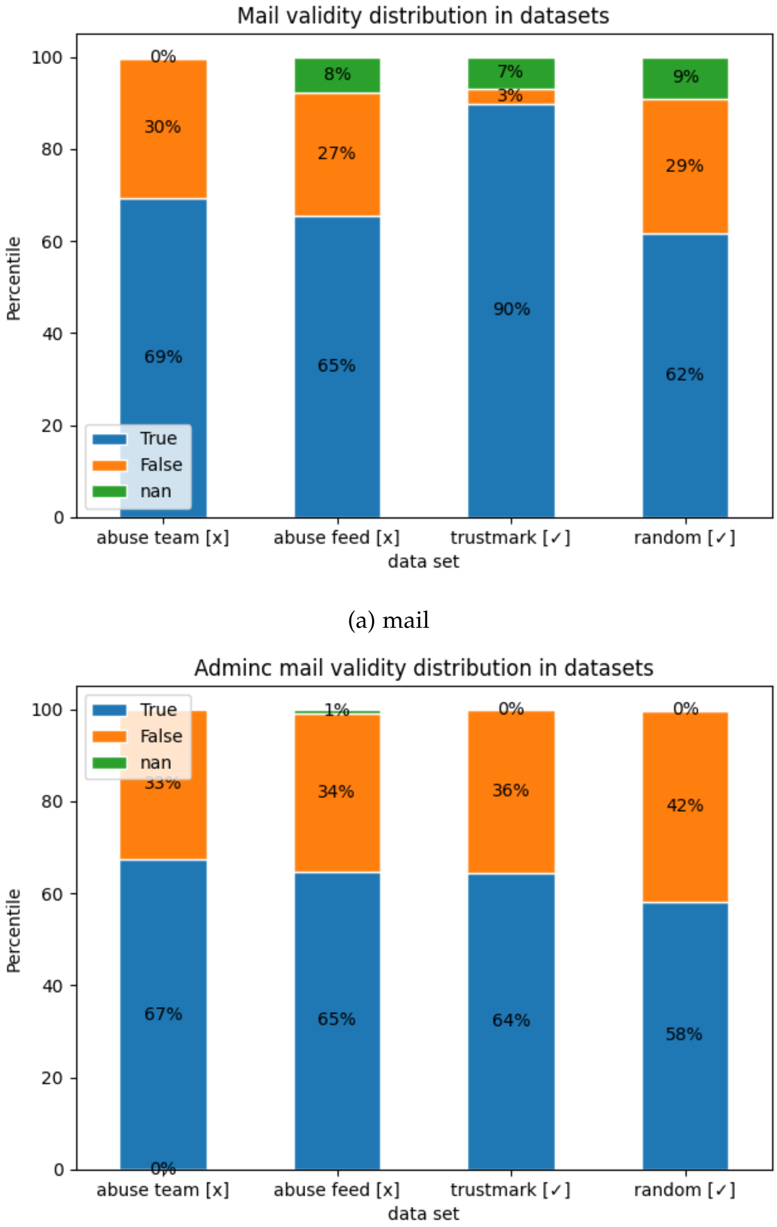
Figure 4.1: Validation results of the name feature over each data set

4.1.2 *E-mail Address*

Figure 4.2 shows the mail validity (as described in Section 3.2.2) and adminc mail validity. A value of **nan** represents a placeholder value of SIDN in their dataset. An interesting statistic is the fact that the mail validity of the normal mail addresses is higher than the mail validity of the adminc mail addresses. It is expected that the adminc mail validity would be higher than the mail validity because the adminc mail addresses are used to contact the domain holder in case something is wrong with the domain. Furthermore, the mail validity for the trustmark data set, such as webshops, is considerably higher while the adminc validity is comparable to the other data sets. Some examples of mail addresses with their validity are given in Table 4.2. The reason the mail addresses *edward.walborn@gmx.com* and *fritchleykysdzr@mynet.com* are invalid is because the recipients *edward.walborn* and *fritchleykysdzr* are not available at the mail servers *gmx.com* and *mynet.com* respectively.

Table 4.2: Examples of the mail feature with their classification result

Mail	Validity
uitverkoop@mindclick.nl	True
support@instalweb.nl	True
edward.walborn@gmx.com	False
fritchleykysdzr@mynet.com	False
gegevens.onbekend@sidn.nl	None



(a) mail

(b) adminc mail

Figure 4.2: Validation results of the mail features over each data set

4.1.3 *Phone Number*

Figure 4.3 shows the phone validity (as described in Section 3.2.3) distribution of the different data sets. the figure shows that the domains that are flagged by the abuse team, in general, have a much higher chance of having a correctly formatted phone number than the other data sets. Some examples of the phone feature with their validity are given in Table 4.3.

Table 4.3: Examples of the phone feature with their classification result

phone	phone_validity
31.0652537096	True
86.59455584700001	False
44.70299942720001	True
3.1638641704	True
49.07044452348	True
45.0036946676	False

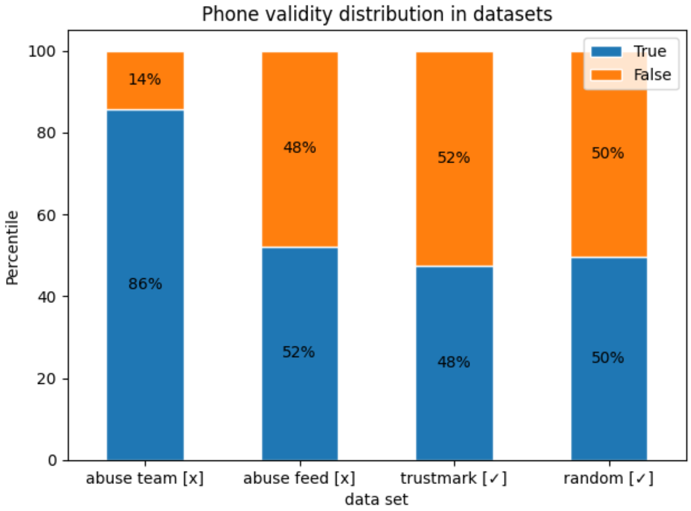


Figure 4.3: validation results of the phone feature over each data set

4.1.4 *Address*

Figure 4.4 shows the address validity (as described in Section 3.2.4) distribution. If the similarity score is between 90 and 100, the domain (almost) exactly matches the domain known in the bag register, while a similarity score between 0 and 40 means that the domains do not match at all. The figure shows that the domains in the abuse team data set, in general, are a lot more likely to be invalid than the domains in the other data sets. Some examples of the Levenshtein distance of the address are given in Table 4.4.

Table 4.4: Examples of the address feature with their Levenshtein distance

address	queried address	validity (Levenshtein distance)
Topaasring 121 5629GE Eindhoven	Topaasring 121 5629GE Eindhoven	100
Tolboomweg 9 3784XC TERSCHUUR	Tolboomweg 9 3784XC Terschuur	100
Eisenhowerstraat 159 1931WL Egmond aan Zee	(no address for the postalcode / street number combination)	0
Wolddijk 1 7981NA Ruinerwold	Wittelterweg 1 7981NA Diever	38

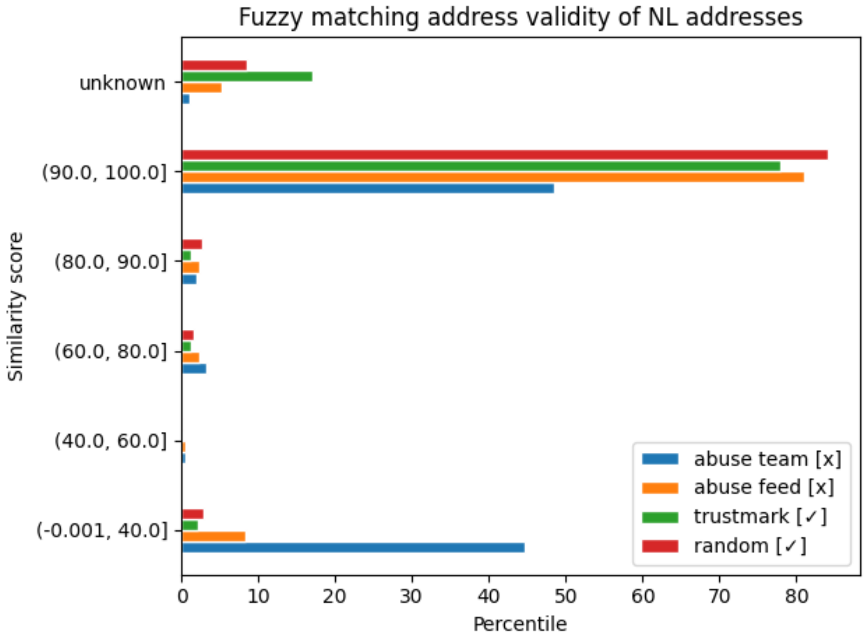


Figure 4.4: validation results of the address feature over each data set

4.1.5 *Combination*

Figure 4.5 shows the combination of the name, address, phone, and email validity. The figure shows the distribution of the number of valid registrations for each data set. Thus, 28% of the random data set has four features that are valid (or correct), whereas the abuse team data set has 8%.

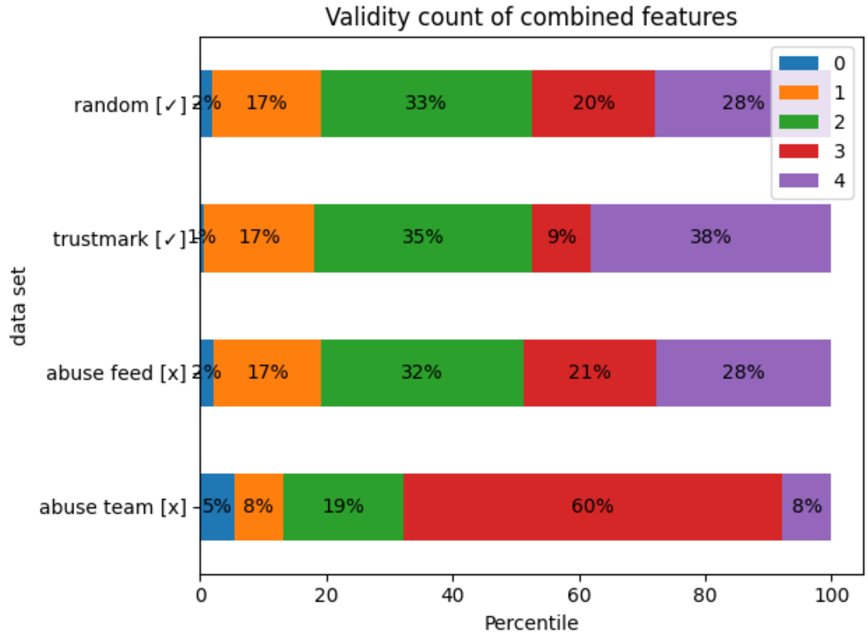


Figure 4.5: Combination results of all the validation features

4.1.6 *Overall Results*

For the phone, mail, address, name, and e-mail address features that were analyzed, the distribution of valid/not valid within each of the four data sets was similar. Furthermore, the distribution of valid/not valid within the malicious data sets (abuse feed, abuse team) is always higher than the distribution within the benign data sets (trustmark, random) except for the adminc mail validity. Also, the distribution of the abuse feed data set and the random data set are for each feature very closely related.

For each data set, the adminc mail validity is worse than that of the normal mail validity.

The fuzzy matching of the addresses in Figure 4.4 shows that more than 90% of each data set falls under two categories, namely 0-40 (completely different) and 90-100 (almost exact match). Furthermore, there is a clear distinction between the abuse team data set and the other data sets.

For the combined features in Figure 4.5, the trustmark data set has a relatively high percentage of 4 (which means every feature is valid). Furthermore, the abuse team data set has a different distribution for the combined features than the other three data sets.

Finally, it is possible to automatically detect false registration information. However, as can be seen in Figure 4.5, the overall validity of normal domains (trustmark and random data sets) is lower than the overall validity of malicious domains (abuse feed and abuse team data sets). Thus, trying to identify malicious domains based on the correctness of the validity of the registration information is not a viable option. Furthermore, the results show that the registration data in general is very untrustworthy, on average, only one out of four registrations contains all valid registration information.

4.2 DATA CLASSIFICATION

This section describes the results of the data classification. Section 4.2.1 reports on the performance of a classifier based on the validity features generated in Section 3.2 to answer the research question *is false registration information a good predictor of the future use of the domain for phishing?* Section 4.2.2 assesses the differently tested classifiers. Further performance improvements with parameter tuning of the random forest classifier are found in Section 4.2.3. The resulting performance metrics of the parameter tuning of the resulting random forest classifier can be found in Section 4.2.4.

4.2.1 *Validation Features Classifier*

From Section 3.2, four different validation features of the registration information were identified and created. To give an indication of their importance for the classification stage, their correlation with the label is tested. This is done by calculating the Pearson correlation coefficient of each feature with regard to the label. The results of the correlation calculation can be seen in Figure 4.6. The higher the correlation with the label attribute, the more important the feature is in classifying registrations with linear classification methods. The figure shows that the mail validity has the least correlation with the label, while the address validity has the most correlation with the label. Furthermore, there are negative correlations between the validity of the address and email address to the label. Showing that a valid address correlates with a malicious domain. Because there is a moderate correlation between each feature and the label, all four features are used in the classification stage.

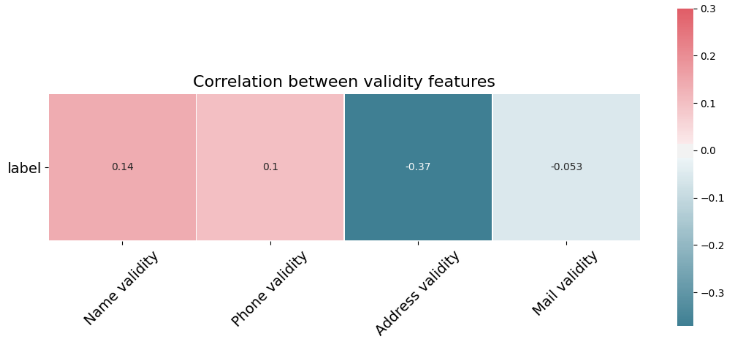


Figure 4.6: Pearson correlation between validation features and the label

After seeing a moderate correlation between the validation features and the label, a Random Forest classifier based on only the validation features was tested. The classifier was trained and tested with 5-fold cross-validation to make sure no bias was introduced based on the split of the data set. The average performance over the five folds was a precision of 0.27, a recall of 0.62, and an F1-score of 0.37.

4.2.2 *Extended Features Classifier*

With the curated data set described in Section 3.3.2, a range of different classifiers was tested. To test the performance of each classifier, 5-fold cross-validation was used to make sure no bias was introduced by simply splitting the data set. The differently tested classifiers and the average performances over the 5 folds can be found in Table 4.5.

Table 4.5: Mean classification metrics of different classifiers

Classifier	Precision	Recall	F1
Random Forest	0.80	0.75	0.77
Logistic Regression	0.24	0.81	0.37
Naïve Bayes	0.05	0.99	0.09
K-nearest neighbours	0.32	0.85	0.46
Decision Tree	0.65	0.71	0.68
Neural Network	0.47	0.82	0.60

From Table 4.5 it follows that the random forest classifier performs the best out of the differently tested classifiers. For this reason, this classifier was used for this research.

4.2.3 *Random Forest Hyper Parameters*

To optimize the random forest classifier, the following hyper parameters with their values were tested:

- The number of trees in the forest (10, 50, 100, 200, 300, 500)
- The minimum number of samples required to be a leaf node (1, 2, 3)
- The minimum number of samples required to split a node (2, 3, 4, 5)

From all the different parameter combinations, the following combination performed the best:

- Number of trees: 100
- samples required to be a leaf node: 1
- samples required to be a split node: 2

4.2.4 *Classifier Results*

Using the random forest classifier with the optimized hyper parameters from Section 4.2.3 the ROC curve, AUC, precision, recall, and f1 metrics were determined. The ROC curve along with the AUC of each fold can be found in Figure 4.7. The average precision, recall, and the f1 score over the 5 folds were 0.80, 0.75, and 0.77 respectively.

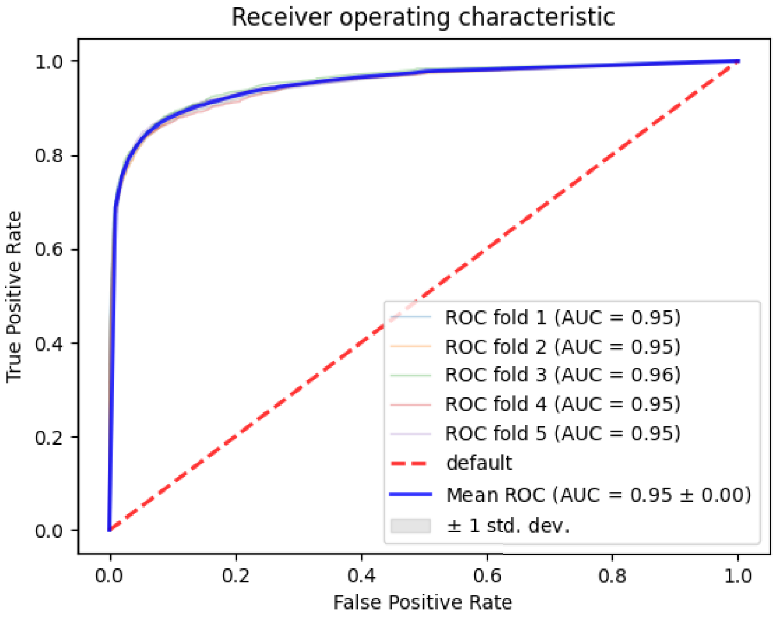


Figure 4.7: ROC curve and AUC with the random forest classifier

Because of the nature of the random forest classifier, the feature importances can be determined. The ten most important features are given in Figure 4.8. Figure 4.8 shows that the count of capital letters in the name feature is the most important feature in the classification of phishing domains.

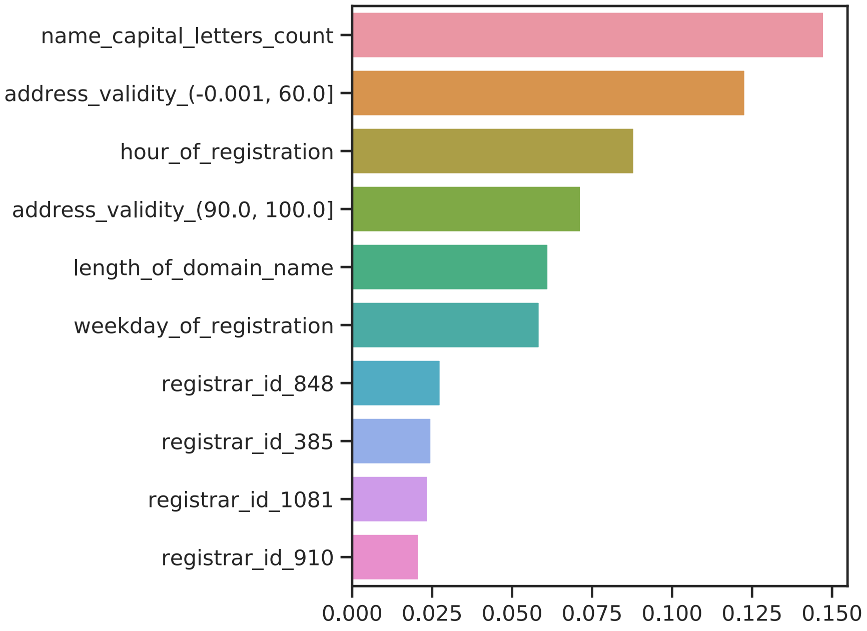


Figure 4.8: Feature importances of the random forest classifier

4.2.5 *Classifier Evaluation*

Using the optimized random forest classifier from Section 4.2.3, the performance on the curated dataset from Section 3.3.6 was tested. As in Section 4.2.4, the ROC curve and AUC are given. Furthermore, because there is no cross-validation on the evaluation data set, the confusion matrix can be given. Table 4.6 gives the confusion matrix, Figure 4.9 shows the ROC curve of the classifier along with the AUC.

Table 4.6: Confusion matrix of the random forest on the evaluation data set

	Predicted benign	Predicted malicious
Actual benign	49897	756
Actual malicious	41	7

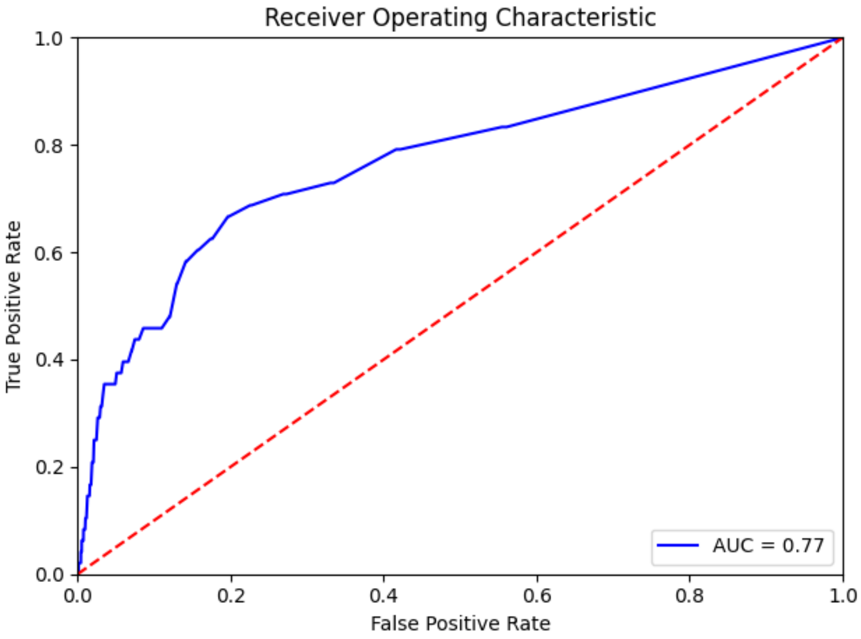


Figure 4.9: ROC curve of the random forest on the evaluation data set

The results show a decline in performance in comparison to the results of the test data set of Section 4.2.4. Because of this, further analysis of the distribution of the features is done.

Figure 4.8 shows that the two most important features of the random forest classifier are: the capital letter count of the name feature and the fuzzy matching address validity between 0 and 60. The comparison of these features between the train and test data set are given in Figure 4.11 and Figure 4.10. Figure 4.10 shows a different distribution between the train and test data. Furthermore, Figure 4.2 shows a drastically different percentage in cases for the address validity between 0 and 60 for the malicious cases between the train and test data set. Additionally, for the malicious domains, there is a high increase in nan labels, meaning that either the address was not formatted correctly at all, or there is a drastic increase in the use of foreign address information with the malicious domains. These distributions result in the decline in classification performance that is seen in Figure 4.9.

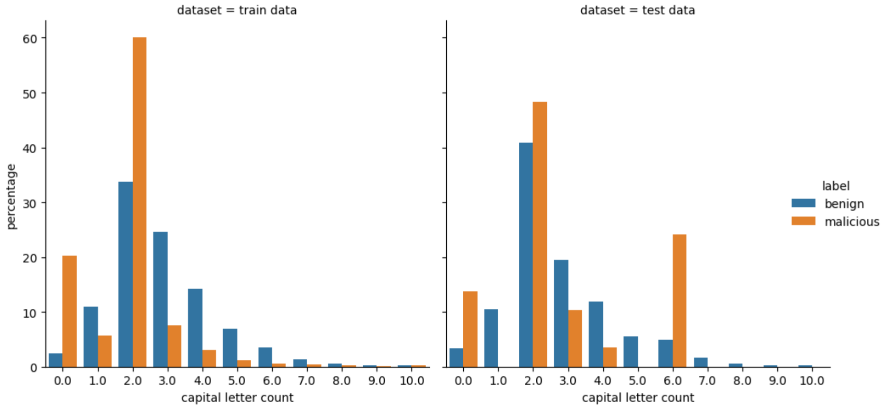


Figure 4.10: Capital letter count distribution of the test and evaluation data sets

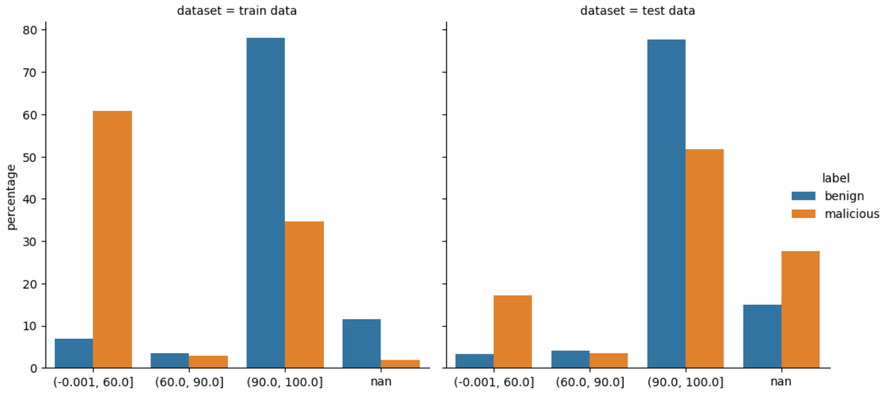


Figure 4.11: Address validity distribution of the test and evaluation data sets

5 DISCUSSION

This chapter discusses the implications of the results in Section 5.1. Furthermore, the limitations of this research are discussed in Section 5.2.

5.1 IMPLICATIONS

The distribution of the percentages of the validity of the four registration features in Section 4.1 showed an exceptionally low overall validity of registration information in the .nl ccTLD. This shows that a large part of the current .nl ccTLD zone has incorrect registration information. This is something that SIDN needs to address. If something is wrong with a domain, SIDN uses this registration information to contact the owner of the domain. Usually, this is done using the administrative mail address. One way to improve this would be to, for example, implement a system where an activation link is sent to the administrative mail address before the domain is added to the zone. This makes sure that all the new registrations, at least at the time of registration, contains a valid mail address. Additionally, SIDN could take a step further and implement a more rigorous approach in the validation of registration information for new domains, like the system that is in place for the .dk ccTLD [15]. This would make it impossible to register new domains with false registration information.

Furthermore, the research shows that using the validity of registration information helps in the prediction of malicious domains. Because of this, existing systems, such as PREMADOMA [41], could integrate these new features to improve the prediction of malicious domains.

For the use of the classifier within SIDN, it can be used as a first filter for the abuse analysts of SIDN. Where the classifier drastically reduces the number of websites the abuse analysts need to go through. Furthermore, the validation system can also be used to aid the abuse analysts by already providing them with the information which registration information features are not correct of suspicious domains. They could use this information to trigger a takedown for these domains. This means that the system does not automatically deny domain registrations.

The results of the study show that it is possible to, with reasonable accuracy, predict which domains become malicious without having to rely on bulk registration features, like previous work [12, 41]. Because the system only utilizes generic registration features, other registries can implementpossible for other registries to implement this system as well.

5.2 LIMITATIONS

Although the research shows interesting results, there are some limitations to the research.

First, the domains in the used data sets were already registered for some time. In this period the registration features, such as the mail address, could have become invalid. This would impact the performance of the resulting classifier in a real-world scenario.

Second, specifically for the validation of the name feature, the registration data set contains a lot of invalid information. In many of the cases where the data says the registration is of a natural person, the registration is actually of a company. As a result, the name validity feature was misclassified many times.

Third, no third-party software or services were used for the validation of personal information. This was specifically chosen not to do due to privacy concerns. However, using third-party services could help in improving the validation performance of the created system.

Fourth, and finally, because this research is publicly available, perpetrators can use the insight information about the setup of this system to avoid detection by the system.

5.3 ETHICAL CONSIDERATIONS

Due to the sensitive nature of the data that was used during this research, measures were taken to ensure that personal information was handled properly. First, a privacy policy analysis was performed for the data collection and processing. The results were reported to the privacy board of SIDN which approved the use of the data as planned in the research. Furthermore, the privacy policy was also approved by the ethics committee of the University of Twente[1]. Second, the created system does not use any online third-party service to process the personal information that is fed into the system.

A second ethical issue for this kind of technology is the risk of discrimination based on origin, religion, gender or race. Within the feature-set that was used to identify malicious domain registrations, the name validity feature may result in such undesired discrimination. This name validity is checked by the use of the natural language processor SpaCy [40]. SpaCy has been trained on a Wikipedia corpus of 2010 [27]. The names appearing in this corpus could not be a good representation of all existing names worldwide. Therefore, there is a risk that people with extraordinary names become flagged more often than the dominant names appearing in the Wikipedia corpus.

[1] Reference number RP 2020-68

The developed system results in a list of domains with a higher risk of being malicious. This list is used as input for an abuse analyst to further check the malicious purpose of the domain. This next step in the process allows for the correction of any unjustified signalling of extraordinary names. The system will not take down domains on its own.

6 CONCLUSION

This research aimed to create a classifier that could predict small scale phishing domains at the time of their registration. By first creating a system that can automatically verify certain parts of the registration information. With this system in place, a classifier was trained with the results of the verification system along with other features available at the time of registration of a domain. The resulting classifier could, with a good performance, classify whether domains would become malicious. This classifier can be used by SIDN to create a first filter where newly registered domains are automatically flagged that are likely going to be used for malicious purposes. Furthermore, the results show that using the validity of the registration information is not a good predictor of malicious intent. To be able to create a classifier that could predict whether a domain would become malicious, two main research questions and five sub-questions were answered.

> **RQ1.1** Which parts of the registration information can be validated and are a useful discriminator to detect false registration data?

Each registration feature was analysed and ways to validate the feature was assessed. From the analysis, it was concluded that four different features of the registration information could be validated and are a useful discriminator to detect false registrations. The identified four features are:

- Phone number
- E-mail address
- Address
- Name

> **RQ1.2** How can the validation of registration information from ***RQ1.1*** be automated?

For each feature that was identified in **RQ1.1**, the possibility of automating the process was researched. Each feature was able to be automated. The results of the process are described in Section 4.1.

> **RQ1** How can false registration information detection be automated?

Combining the results of the previously described sub-questions, the first main research question can be answered. Using the process of the automation of the four identified registration features described in Section 3.2, a system can be created to flag registrations that have false registration features. This system is used in the classification part of the research as input features for the classifiers.

> **RQ2.1** Is false registration information a good predictor of the future use of the domain for phishing?

By curating a data set with only features of the system that was created in **RQ1**, a classifier was trained and tested. The results are presented in Section 4.2.1. From these results, it can be concluded that a classifier based on only the validation features is not sufficient. However, from the feature importances given in Figure 4.8, the validation features do contribute to a good classification performance.

> **RQ2.2** What other features are good predictors of the future use of a domain for phishing?

A combination of features of previous research was determined as good predictors. These additional features are described in Section 3.3.2.

> **RQ2.3** What is the detection performance of phishing domains in the .nl top-level domain of a classifier based on the features identified in ***RQ2.1*** and ***RQ2.2***?

Using a combination of the identified features of **RQ2.1** and **RQ2.2**, different classifiers were tested. The results of these classifiers are reported in Section 4.2.2. From the different tested classifiers, the random forest classifier performed best. Further performance improvements were gained by tuning the hyperparameters of the random forest classifier. Section 4.2.3 reports on the results of the hyperparameter tuning. Using the tuned random forest classifier, the final prediction performance of the classifier was tested. Table 4.5 describes the final performance of the tuned random forest classifier, namely, a precision of 0.80, a recall of 0.75, and an f1 score of 0.77.

> **RQ2.4** What are the best model parameters such that the classifier of ***RQ2.3*** performs the best according to SIDN?

The main purpose of the created classifier of **RQ2.3** for SIDN is a tool that can filter all new registrations and create a list in which registrations that look suspicious are documented. This is useful for the abuse team of SIDN to speed up the process of the fight against abuse. Because of this purpose, the classifier was trained on a higher precision.

> **RQ2** Can small scale phishing domains be detected at time of registration for second-level domains in the .nl top-level domain?

Combining the results of the previous sub-questions, it can be concluded that it is possible to detect phishing domains at the time of registration. The results of this research can be used by SIDN to further improve the impact their abuse team have on keeping the .nl ccTLD safe.

6.1 FUTURE WORK

This research revealed multiple routes for future work. First, by expanding the currently used feature set with additional features that are available at the time of registration to improve classification performance. Further improvements could be made by implementing alternative machine learning algorithms, such as using an active learning strategy in harmony with the abuse team of SIDN to improve the classification performance continuously. Additionally, the bulk classification features available in PREMADOMA [41], and this research, can be combined into a classifier to further improve detection performance.

Second, by testing the developed system in this research in multiple different TLDs, a better insight into the real-world performance of the system can be made. As well as research whether such a system would perform on the registrar level, instead of at the registry level.

A APPENDIX

A.1 ABUSE TOKENS

```
login
rekening
marktplaats
helpdesk
iban
controle
zorgverzekeringen
hacker
rabobank
bankieren
inloggen
log
hotmail
pay
betaal
verifieren
lng
aanvraag
verificatie
vervanging
abn
pas
rabo
instagram
snapchat
secure
klanten
account
intern
scanner
verzoek
procedure
amro
vernieuwde
omgeving
melding
vodafone
vervangen
apple
sns
paypal
gegevens
updates
```

```
update
portaal
vervang
beveiligd
twitter
upgrade
klant
access
meldingen
nfc
card
controleer
formulier
koppeling
transacties
ziggo
identificatie
controleren
proces
blokkeren
netflix
banking
betaling
pass
bing
```

A.2 ERRATUM

After the publication of the thesis, SIDN verified the address validation algorithm as developed in this thesis and described in Section 4.1. During this verification, SIDN used the same method and algorithm for the address validation check as described in Section 3.2.

For the verification, SIDN compiled a new random sample set of 100 000 domains with unique addresses and at least 31 days after registration. The set was split into two parts based on whether the domain could be found on an abuse list. 270 of the 100 000 domains were found on an abuse list.

The test results show that 3.5% of the benign domains had an address validity score of < 40. The mean address validity score of the benign domains was 94. Furthermore, 28.3% of the malicious domains had an address validity score of < 40 with a mean score of 75.

These tests show that there is a difference in address validity between malicious and benign domains, which contradicts the results found in Section 4.1. These initial results showed no significant difference between the address validity of malicious domains and benign domains.

It proved impossible to identify a clear explanation for the results of the initial set of tests, but the re-evaluation by SIDN shows that somehow the dataset used for the initial tests contained errors. The conclusions presented in Section 4.1 regarding the address validity of benign domains are likely incorrect.

The results of the evaluation of this new random sample set improve the performance of a classification algorithm to distinguish between malicious and benign domains, because there appears to be a clear difference in the data between the address validity of a benign domain and a malicious domain. This is expected to lead to a better classification performance than described in this thesis in Section 4.2.2.

[1] Maher Aburrous, M. A. Hossain, Fadi Thabatah, and Keshav Dahal. "Intelligent Phishing website detection system using Fuzzy techniques." In: *2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications, ICTTA* (2008), pp. 1–6. DOI: 10.1109/ICTTA.2008.4530019.

[2] Khulood Al Messabi, Monther Aldwairi, Ayesha Al Yousif, Anoud Thoban, and Fatna Belqasmi. "Malware detection using DNS records and domain name features." In: *ACM International Conference Proceeding Series* (2018). DOI: 10.1145/3231053.3231082.

[3] *APWG | Unifying The Global Response To Cybercrime*. URL: https://apwg.org/ (visited on 05/28/2020).

[4] Sushma Nagesh Bannur, Lawrence K. Saul, and Stefan Savage. "Judging a site by its content: Learning the textual, structural, and visual features of malicious web pages." In: *Proceedings of the ACM Conference on Computer and Communications Security* (2011), pp. 1–9. ISSN: 15437221. DOI: 10.1145/2046684.2046686.

[5] Rok Blagus and Lara Lusa. "SMOTE for high-dimensional class-imbalanced data." In: *BMC Bioinformatics* 14.1 (Dec. 2013), p. 106. ISSN: 1471-2105. DOI: 10.1186/1471-2105-14-106. URL: https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-14-106.

[6] Davide Canali, Marco Cova, Giovanni Vigna, and Christopher Kruegel. "Prophiler: A fast filter for the large-scale detection of malicious web pages." In: *Proceedings of the 20th International Conference on World Wide Web, WWW 2011* (2011), pp. 197–206. DOI: 10.1145/1963405.1963436.

[7] *Ceo-fraude kostte Pathé 19 miljoen euro | Het Parool*. URL: https://www.parool.nl/nieuws/ceo-fraude-kostte-pathe-19-miljoen-euro%7B~%7Db8ae182c/?referer=https%7B%5C%%7D3A%7B%5C%%7D2F%7B%5C%%7D2Fwww.google.com%7B%5C%%7D2F (visited on 03/09/2020).

[8] Daiki Chiba, Kazuhiro Tobe, Tatsuya Mori, and Shigeki Goto. "Detecting malicious websites by learning IP address features." In: *Proceedings - 2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet, SAINT 2012* (2012), pp. 29–39. DOI: 10.1109/SAINT.2012.14.

[9] *General Terms and Conditions for .nl Registrants*. Tech. rep. 2019. URL: https://www.sidn.nl/downloads/d%7B%5C_%7D7zdiiDQvOGbSo1FGCcqw/d4c8288846f98ba422834c996994a04a/General%7B%5C_%7DTerms%7B%5C_%7Dand%7B%5C_%7DConditions%7B%5C_%7Dfor%7B%5C_%7Dnl%7B%5C_%7DRegistrants.pdf.

[10] *google/libphonenumber: Google's common Java, C++ and JavaScript library for parsing, formatting, and validating international phone numbers*. URL: https://github.com/google/libphonenumber (visited on 06/01/2020).

[11] Sachin Gupta. "Efficient malicious domain detection using word segmentation and BM pattern matching." In: *2016 International Conference on Recent Advances and Innovations in Engineering, ICRAIE 2016* (2016), pp. 1–6. DOI: 10.1109/ICRAIE.2016.7939534.

[12] Shuang Hao, Alex Kantchelian, Brad Miller, Vern Paxson, and Nick Feamster. "PREDATOR: Proactive recognition and elimination of domain abuse at time-of-registration." In: *Proceedings of the ACM Conference on Computer and Communications Security*. Vol. 24-28-Octo. 2016, pp. 1568–1579. ISBN: 9781450341394. DOI: 10.1145/2976749.2978317.

[13] Shuang Hao, Matthew Thomas, Vern Paxson, Nick Feamster, Christian Kreibich, Chris Grier, and Scott Hollenbeck. "Understanding the domain registration behavior of spammers." In: *Proceedings of the ACM SIGCOMM Internet Measurement Conference, IMC* (2013), pp. 63–75. DOI: 10.1145/2504730.2504753.

[14] *Home - Webshops - Webshop Keurmerk*. URL: https://www.keurmerk.info/en/home-en/ (visited on 07/28/2020).

[15] *ID check | DK Hostmaster*. URL: https://www.dk-hostmaster.dk/en/id-check (visited on 10/01/2020).

[16] *Internet Crime Complaint Center (IC3) | Business E-mail Compromise The 12 Billion Dollar Scam*. URL: https://www.ic3.gov/media/2018/180712.aspx (visited on 12/30/2019).

[17] S. Carolin Jeeva and Elijah Blessing Rajsingh. "Intelligent phishing url detection using association rule mining." In: *Human-centric Computing and Information Sciences* 6.1 (2016). ISSN: 21921962. DOI: 10.1186/s13673-016-0064-3.

[18] H. B. Kazemian and S. Ahmed. "Comparisons of machine learning techniques for detecting malicious webpages." In: *Expert Systems with Applications* 42.3 (2015), pp. 1166–1177. ISSN: 09574174. DOI: 10.1016/j.eswa.2014.08.046. URL: http://dx.doi.org/10.1016/j.eswa.2014.08.046.

[19] Firoz Khan, Jinesh Ahamed, Seifedine Kadry, and Lakshmana Kumar Ramasamy. “Detecting malicious URLs using binary classification through ada boost algorithm.” In: *International Journal of Electrical and Computer Engineering* 10.1 (2020), pp. 997–1005. ISSN: 20888708. DOI: 10.11591/ijece.v10i1.pp997-1005.

[20] Barbara Kitchenham. “Procedures for Performing Systematic Literature Reviews.” In: *Joint Technical Report, Keele University TR/SE-0401 and NICTA TR-0400011T.1* (2004), p. 33.

[21] Rajesh Kumar, Xiaosong Zhang, Hussain Ahmad Tariq, and Riaz Ullah Khan. “Malicious URL detection using multi-layer filtering model.” In: *2016 13th International Computer Conference on Wavelet Active Media Technology and Information Processing, ICCWAMTIP 2017* 2018-Febru (2017), pp. 97–100. DOI: 10.1109/ICCWAMTIP.2017.8301457.

[22] *Large-scale phishing campaign targets Canadian banks*. URL: https://www.scmagazineuk.com/large-scale-phishing-campaign-targets-canadian-banks/article/1669605 (visited on 04/21/2020).

[23] Tie Li, Gang Kou, and Yi Peng. “Improving malicious URLs detection via feature engineering: Linear and nonlinear space transformation methods.” In: *Information Systems* 91 (2020), p. 101494. ISSN: 0306-4379. DOI: 10.1016/j.is.2020.101494. URL: https://www.sciencedirect.com/science/article/pii/S0306437920300053?dgcid=rss%7B%5C_%7Dsd%7B%5C_%7Dall%7B%5C&%7Dutm%7B%5C_%7Dsource=researcher%7B%5C_%7Dapp%7B%5C&%7Dutm%7B%5C_%7Dmedium=referral%7B%5C&%7Dutm%7B%5C_%7Dcampaign=RESR%7B%5C_%7DMRKT%7B%5C_%7DResearcher%7B%5C_%7Dinbound.

[24] Justin Ma, Lawrence K. Saul, Stefan Savage, and Geoffrey M. Voelker. “Beyond blacklists: Learning to detect malicious web sites from suspicious URLs.” In: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2009), pp. 1245–1253. DOI: 10.1145/1557019.1557153.

[25] Asha S Manek, V Sumithra, P Deepa Shenoy, M. Chandra Mohan, K R Venugopal, and L M Patnaik. “DeMalFier: Detection of Malicious web pages using an effective classifier.” In: *2014 International Conference on Data Science & Engineering (ICDSE)*. IEEE, Aug. 2014, pp. 83–88. ISBN: 978-1-4799-5460-5. DOI: 10.1109/ICDSE.2014.6974616. URL: http://ieeexplore.ieee.org/document/6974616/.

[26] Alexander Moshchuk, Tanya Bragin, Damien Deville, Steven D. Gribble, and Henry M. Levy. “SpyProxy: Execution-based detection of malicious web content.” In: *16th USENIX Security Symposium* (2007), pp. 27–42.

[27] *Multi-language · spaCy Models Documentation*. URL: https://spacy.io/models/xx (visited on 06/01/2020).

[28] K. Nandhini and Ramesh Balasubramaniam. "Malicious Website Detection Using Probabilistic Data Structure Bloom Filter." In: *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*. IEEE, Mar. 2019, pp. 311–316. ISBN: 978-1-5386-7808-4. DOI: 10.1109/ICCMC.2019.8819818. URL: https://ieeexplore.ieee.org/document/8819818/.

[29] NCTV. "Cybersecuritybeeld nederland." In: *Nctv* (2019), pp. 1–76. URL: https://www.thehaguesecuritydelta.com/media/com%7B%5C_%7Dhsd/report/237/document/CSBN2019-online-tcm31-392768.pdf.

[30] *numverify API | Free Phone Number Validation & Lookup API*. URL: https://numverify.com/ (visited on 06/05/2020).

[31] *OpenStreetMap*. URL: https://www.openstreetmap.org/ (visited on 06/05/2020).

[32] Yongfang Peng, Shengwei Tian, Long Yu, Yalong Lv, and Ruijin Wang. "Malicious URL recognition and detection using attention-based CNN-LSTM." In: *KSII Transactions on Internet and Information Systems* 13.11 (2019), pp. 5580–5593. ISSN: 22881468. DOI: 10.3837/tiis.2019.11.017.

[33] *PhishTank | Join the fight against phishing*. URL: https://www.phishtank.com/ (visited on 05/28/2020).

[34] G. Pranav Naidu and K. Govinda. "Bankruptcy prediction using neural networks." In: *Proceedings of the 2nd International Conference on Inventive Systems and Control, ICISC 2018* 11 (2018), pp. 248–251. DOI: 10.1109/ICISC.2018.8399072.

[35] Kotoju Rajitha and Doddapaneni VijayaLakshmi. "Oppositional Cuckoo Search Based Weighted Fuzzy Rule System in Malicious Web Sites Detection from Suspicious URLs." In: *International Journal of Intelligent Engineering and Systems* 9.4 (2016), pp. 116–125. ISSN: 21853118. DOI: 10.22266/ijies2016.1231.13.

[36] Routhu Srinivasa Rao and Alwyn Roshan Pais. "Detection of phishing websites using an efficient feature-based machine learning framework." In: *Neural Computing and Applications* 31.8 (Aug. 2019), pp. 3851–3873. ISSN: 14333058. DOI: 10.1007/s00521-017-3305-0.

[37] *Safe Browsing – Google Safe Browsing*. URL: https://safebrowsing.google.com/ (visited on 02/25/2020).

[38] Shymala Gowri Selvaganapathy, Mathappan Nivaashini, and Hema Priya Natarajan. "Deep belief network based detection and categorization of malicious URLs." In: *Information Security Journal* 27.3 (2018), pp. 145–161. ISSN: 19393547. DOI: 10.1080/19393555.2018.1456577. URL: https://doi.org/10.1080/19393555.2018.1456577.

[39] *SmartScreen: FAQ*. URL: https://support.microsoft.com/en-us/help/17443/microsoft-edge-smartscreen-faq (visited on 02/25/2020).

[40] *spaCy · Industrial-strength Natural Language Processing in Python*. URL: https://spacy.io/ (visited on 06/01/2020).

[41] Jan Spooren, Thomas Vissers, Peter Janssen, Wouter Joosen, and Lieven Desmet. "Premadoma: An operational solution for DNS registries to prevent malicious domain registrations." In: *ACM International Conference Proceeding Series* (2019), pp. 557–567. DOI: 10.1145/3359789.3359836.

[42] Ivan Torroledo, Luis David Camacho, and Alejandro Correa Bahnsen. "Hunting malicious tls certificates with deep neural networks." In: *Proceedings of the ACM Conference on Computer and Communications Security* (2018), pp. 64–73. ISSN: 15437221. DOI: 10.1145/3270101.3270105.

[43] *URLhaus | Malware URL exchange*. URL: https://urlhaus.abuse.ch/ (visited on 05/28/2020).

[44] *Verificatie registratiegegevens | SIDN*. URL: https://www.sidn.nl/nl-domeinnaam/verificatie-registratiegegevens (visited on 03/12/2020).

[45] Thomas Vissers, Jan Spooren, Pieter Agten, Dirk Jumpertz, Peter Janssen, Marc Van Wesemael, Frank Piessens, Wouter Joosen, and Lieven Desmet. "Exploring the Ecosystem of Malicious Domain Registrations in the.eu TLD." In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10453 LNCS (2017), pp. 472–493. ISSN: 16113349. DOI: 10.1007/978-3-319-66332-6_21.

[46] Guang Xiang, Jason Hong, Carolyn P Rose, and Lorrie Cranor. "CANTINA+." In: *ACM Transactions on Information and System Security* 14.2 (Sept. 2011), pp. 1–28. ISSN: 10949224. DOI: 10.1145/2019599.2019606. URL: http://dl.acm.org/citation.cfm?doid=2019599.2019606.

[47] Cuiwen Xiong, Pengxiao Li, Peng Zhang, Qingyun Liu, and Jianlong Tan. "MIRD: Trigram-based malicious URL detection implanted with random domain name recognition." In: *Communications in Computer and Information Science* 557 (2015), pp. 303–314. ISSN: 18650929. DOI: 10.1007/978-3-662-48683-2_27.

[48] Wenchuan Yang, Wen Zuo, and Baojiang Cui. "Detecting Malicious URLs via a Keyword-Based Convolutional Gated-Recurrent-Unit Neural Network." In: *IEEE Access* 7 (2019), pp. 29891–29900. ISSN: 21693536. DOI: 10.1109/ACCESS.2019.2895751. URL: https://ieeexplore.ieee.org/document/8629082/.

[49] Wen Zhang, Yu Xin Ding, Yan Tang, and Bin Zhao. "Malicious web page detection based on on-line learning algorithm." In: *Proceedings - International Conference on Machine Learning and Cybernetics* 4 (2011), pp. 1914–1919. ISSN: 2160133X. DOI: 10.1109/ICMLC.2011.6016954.

[50] Yue Zhang, Jason I. Hong, and Lorrie F. Cranor. "Cantina." In: *Proceedings of the 16th international conference on World Wide Web - WWW '07*. New York, New York, USA: ACM Press, 2007, p. 639. ISBN: 9781595936547. DOI: 10.1145/1242572.1242659. URL: http://portal.acm.org/citation.cfm?doid=1242572.1242659.

[51] Peilin Zhao and Steven C.H. Hoi. "Cost-sensitive online active learning with application to malicious URL detection." In: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* Part F1288 (2013), pp. 919–927. DOI: 10.1145/2487575.2487647.