applied sciences

MDPI

Article

Experimental Evaluation of Possible Feature Combinations for the Detection of Fraudulent Online Shops

Audronė Janavičiūtė, Agnius Liutkevičius *, Gedas Dabužinskas and Nerijus Morkevičius

Department of Computer Sciences, Kaunas University of Technology, 44249 Kaunas, Lithuania; audrone.janaviciute@ktu.lt (A.J.); gedas.dabuzinskas@ktu.edu (G.D.); nerijus.morkevicius@ktu.lt (N.M.)
* Correspondence: agnius.liutkevicius@ktu.lt

Abstract: Online shopping has become a common and popular form of shopping, so online attackers try to extract money from customers by creating online shops whose purpose is to compel the buyer to disclose credit card details or to pay money for goods that are never delivered. Existing buyer protection methods are based on the analysis of the content of the online shop, customer reviews, the URL (Uniform Resource Locator) of the website, the search in blacklists or whitelists, or the combination of the above-mentioned methods. This study aims to find the minimal set of publicly and easily obtainable features to create high-precision classification solutions that require little computing and memory resources. We evaluate various combinations of 18 features that belong to three possible categories, namely URL-based, content-based, and third-party services-based. For this purpose, the custom dataset is created, and several machine learning models are applied for the detection of fraudulent online shops based on these combinations of features. The results of this study show that even only four of the most significant features allow one to achieve 0.9342 classification accuracy, while 0.9605 accuracy is reached with seven features, and the best accuracy of 0.9693 is achieved using thirteen and fifteen features.

Keywords: fake-shop detection; E-commerce; fraud detection; machine learning; cybersecurity

Citation: Janavičiūtė, A.; Liutkevičius, A.; Dabužinskas, G.; Morkevičius, N. Experimental Evaluation of Possible Feature Combinations for the Detection of Fraudulent Online Shops. *Appl. Sci.* **2024**, *14*, 919. https://doi.org/10.3390/app14020919

Academic Editor: Gianluca Lax

Received: 27 December 2023
Revised: 16 January 2024
Accepted: 18 January 2024
Published: 22 January 2024

1. Introduction

As more people and businesses conduct various personal and commercial activities online, the risk of becoming a victim of cybercrime and, in particular, fraud is very high. According to recent surveys [1,2], there are several types of financial fraud, including but not limited to credit card fraud, insurance fraud, money laundering, fraudulent financial transactions, financial statement fraud, cryptocurrency fraud, healthcare fraud, and securities and commodities fraud. There are many research studies related to the problem of fraud detection as reviewed in refs. [1,2]. Fraud can be implemented using a variety of tactics, including phishing. Phishing is a network attack that combines social engineering and computer technology to steal sensitive personal information from users [3]. Usually, attackers create phishing websites that look like legitimate ones to trick people into clicking links to malicious websites and submitting sensitive information, for example, account credentials. There are many studies that propose methods to detect phishing websites, as surveyed in refs. [3,4]. However, few studies have explored the problem of fake online shops. Fraudulent online shops are the case of phishing websites that pretend to be legitimate. Many people in the world use online shops every day. According to Statista [5] about 2.14 billion people shopped online in 2021 and this number is growing constantly. It presents a good opportunity for attackers to exploit the naiveness and recklessness of buyers, offering fake online shops to obtain personal data, credit card data, or money from buyers.

Scammers create stores that closely resemble real or existing stores of well-known brands. In this way, the buyer is made to believe that he or she is shopping at a legitimate

and reliable store and, without hesitation, provides his or her personal data, credit card data, and pays for the goods. The buyer realizes the fraud when he or she does not receive the goods or sees that money has disappeared from his or her credit card account. According to Statista [6], 75% of consumers who were victims of online purchase scams worldwide lost money in 2021, while the median monetary loss per online purchase scam reached 101 US dollars [7].

Detecting scam online shops is typically based on website content and URL (Uniform Resource Locator), blacklists or whitelists, consumer reviews and complaints. Most researchers propose methods to detect scam websites using publicly available general data sets, which include phishing websites, fake news, piracy websites, etc. The most popular datasets used for the detection of phishing websites using machine learning techniques are PhishTank [8], Alexa [9], UCI Machine Learning Repository [10], OpenPhish [11], Common Crawl [12], and ISCXURL-2016 [13]. The machine learning approach is most common for the analysis of website content and address. Content analysis is based on the features related to HTML (Hyper Text Markup Language) [14–16], body text [17], images [17–20], domain registration information [15,21] and styles [22]. The URL analysis [21,23–27] is usually based on the features related to the length of the full URL, the domain name, the directory, the types of symbols, the protocol, the number of symbols, etc. Another popular way to assess whether a website is malicious is through various specialized portals that collect data on various fraudulent websites reported by customers. Popular platforms for collecting malicious websites, such as PhishTank [8], Alexa [9], ScamAdviser [28], URLVoid [29] or VirusTotal [30], have accumulated millions of entries in their blacklists, which allows users to judge whether a website is legitimate. Furthermore, various rating and review platforms, such as TrustPilot [31] can be used to assess the trustworthiness of websites, indicating a potentially malicious website if it has a lot of negative feedback.

The analysis of the scientific literature shows that researchers use many types of features for the detection of fraudulent websites (we found more than 100 unique features in the scientific literature, e.g., [23]). A significant number of these features are difficult to obtain and require complex and time-consuming calculations, for example [32]. Classification models based on these features are often large-scale, and their application in practice requires large computing and memory resources. The aim of our work is to determine which features are best suited to detect fraudulent online shops. The goal is to find the smallest possible set of features that can be used to create a fast and small-scale classification solution based on them. To reduce the amount of computations, an initial set of only 18 features is chosen including features that can be extracted directly from the store's URL and its website source code by performing a simple text search. Another part of this set consists of features, which are obtained directly from third parties without requiring additional complex data processing. We construct different combinations of these features and evaluate their suitability for identifying fraudulent online shops by creating machine learning-based classifiers and evaluating their detection accuracy. The combinations of features of different sizes and compositions are used to evaluate the classification accuracy in order to find the most important features, giving the highest possible accuracy. Most of the features are obtained from public services such as TrustPilot [31], WHOIS [33], Tranco list [34], and Sitejabber [35] while the content-based features include favicon, payment methods, and contact details. Additionally, we try to use URL-based features that are specific to online shops only, excluding those that are more suitable for phishing websites such as files, parameters, and queries. For this study, we created our own dataset that contains 1140 records of fraudulent and legitimate online shops using publicly available sources [36]. Publicly available datasets such as PhishTank [8] and VirusTotal [30] are not suitable for our research purposes, because they contain not only malicious online stores, but all kinds of fraudulent websites, including phishing, e-commerce, fake news, piracy, etc.

To our knowledge, there are few scientific works that deal only with the identification of fraudulent online shops compared to methods for identifying any kind of malicious

websites. Shin et al. [32] proposed the method for real-time detection of fake e-commerce sites based on the similarity of the DOM (Document Object Model) trees of websites reaching 0.998 accuracy using a SVM (Support Vector Machine) classifier trained with 3597 fake sites reported by users and blacklisted by officials. Khoo et al. [17] presented another content-based website classification method. This method allows detection of fraudulent e-commerce websites based on three types of features, namely HTML tags, textual content and image of the website, and achieved 0.987 accuracy using Linear Regression as a classifier. Beltzung et al. [37] proposed an approach to classify fraudulent online shops solely on the basis of the similarity of their source code structure using machine learning processes. They used features including tokenized HTML, CSS and JavaScript text, comments and individual tags, tag-attribute-value patterns, and the HTML tree structure, reaching the accuracy of up to 0.97 and a very high degree of certainty in classifying fraudulent e-commerce. Sánchez-Paniagua et al. [38] proposed to use features obtained from third-party sources, including Trustpilot [31] and WHOIS [33] together with features obtained from the website source and metadata. These features include high discounts, social media footprint, domain age, registration date, SSL names, country and issuer, Trustpilot score and review, e-commerce technologies and policies. This method achieved a 0.75 F1-score and 0.86 accuracy when using the Random Forest classifier.

Our proposed approach differs from the ones dedicated to the detection of fraudulent online shops in that it tries to find the most important features, which have the greatest impact on the classification accuracy. At the same time, we try to find the least number of features possible that would guarantee high accuracy but that can be easily obtained and do not require complex computations and content analysis. Such a minimal feature set can be useful for various practical applications like browser add-ons for the detection of fake online shops and other anti-malware software, which do not require complex computations and webpage analysis and consider only publicly available data. The results of this study show that 0.9605 classification accuracy is achieved only with 7 features including *F15—Indication of young domain*, *F9—Presence of money back payment*, *F2—Top domain length*, *F16—Presence of TrustPilot reviews*, *F13—Presence of logo URL, F7—Number of hyphens (-)*, and *F3Presence of prefix "www"*. The best classification accuracy of 0.9693 is achieved with a combination of 13 features.

This article is organized as follows. Section 2 presents the methodology for the evaluation of possible feature combinations, describes the proposed features, and describes the experimental setup. Section 3 shows the results of the evaluation and presents the classification accuracies of the best feature combinations. Section 4 discusses the results and compares them with previous research. Section 5 concludes the study.

2. Materials and Methods

2.1. Methodology

In this work, we used a machine learning-based approach for the detection of fraudulent online shops based on different combinations of features, including URL content (length, symbol count), website content (favicon, payment methods, contact email), public reviews and rankings, domain age, and organization of the issuer of the SSL certificate. The primary dataset that contains all 18 features was created using various data sources.

Because one of the goals was to find the most significant features, we constructed feature subsets of different lengths that include a portion of the initial set of 18 features. For this purpose, we used combinations of features. Mathematically speaking, a *combination* is a selection of items from a set that has distinct members, such that the order of selection does not matter. Based on the combinatorial formula for calculating combinations, the number of these combinations is equal to $C(n, m) = n!/(m!(n\text{-}m)!)$, where n is the number of all features and m is the size of the feature subset. For the number of features $m = 1...5$ and $m = 14...18$, all possible combinations $C_{i,m}$ ($i = 1...C(n,m)$) of the secondary datasets were created and classified using traditional machine learning algorithms. After evaluating the classification results, one the most significant feature (*F15—Indication of young domain*)

was included in all combinations, and one of the least significant features (*F18—Presence in the standard Tranco list*) was excluded from all combinations where $m = 6...13$. Therefore, 16 features instead of 18 were used to create combinations for the secondary datasets, where $m = 6...13$. This was done to reduce the number of calculations that was quite high for certain values of n and m (e.g., $C(18,11) \sim 201 \times 10^9$). For each combination of features $C_{i,m}$, where $i = 1...C(n, m)$, secondary datasets were created and processed by applying seven traditional machine learning algorithms. Each secondary dataset was divided into training and testing sets, applying different splitting strategies. There were 4 variants of splitting: 80/20, 60/40, 40/60 and 20/80 (e.g., 80/20 means 80% of data was used for training and 20% of data was used for testing). All splitting variants were applied for combinations where $m = 1...5$ and $m = 14...18$. Initial experiments showed that the best results were obtained with the 80/20 split, which was used for the remaining combinations of features ($m = 6...13$). Training and testing sets were used to train and evaluate machine learning classifiers, namely Decision Tree, Random Forest, Stochastic Gradient Descent, Logistic Regression, Gaussian Naive Bayes, Multilayer Perceptron, and XGBoost. The reason for the selection of these classification models was their diversity and frequent use in previous studies demonstrating good classification results [17,37,38].

The final step of this methodology was to evaluate the classification results. We used accuracy as the main evaluation metric, which is described using a common equation [39]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

where *TP*—True Positive, *TN*—True Negative, *FP*—False Positive and *FN*—False Negative. The meanings of the components of Equation (1) are given below:

- True Positive (*TP*). The number of cases where an online shop is correctly classified as fraudulent.
- False Positive (*FP*). The number of cases where an online shop is incorrectly classified as fraudulent.
- True Negative (*TN*). The number of cases where an online shop is correctly identified as legitimate.
- False Negative (*FN*). The number of cases where an online shop is incorrectly classified as legitimate.

The flowchart of the proposed methodology is shown in Figure 1.

2.2. Primary Dataset Preparation

In this study, we used publicly available sources to create our own dataset [36] that contains 1140 records of 579 fraudulent and 561 legitimate online shops. This dataset consists of 18 features obtained by analyzing the URL of the online shop, checking the SSL certificate, analyzing the content of the online shop, and receiving features from third-party publicly available services. This custom dataset was created for the following reasons:

1. The majority of the publicly available datasets contain all kinds of phishing websites, characterized by different features, some of them completely not relevant to online shops.
2. The existing datasets dedicated to online shops do not contain all the proposed features, which require one to extract additional data from the website content and third-party services.

In order to create the initial URL list of fraudulent online shops (see Figure 1), a few publicly available sources were used: Watchlist Internet [40], Artists Against 419 [41], and Global e-Commerce Websites List [42]. This list was complemented with URLs of legitimate online shops taken from Trusted shops [43], Ecommerce Trustmark [44], EHI Geprüfter Online-Shop [45], Retail Excellence [46] and Similarweb [47], and manually adding URLs of well-known shops. As a result, we obtained a file with records containing the URL of the web shop and its label (malware or safe). These URLs were used to create the

primary dataset using a Python script and manual procedures to extract the 18 features. The Python script was used to analyze the URL of the online shop, the SSL certificate, and the source code of its website. This script also called APIs of third-party services, including WHOIS, TrancoList, TrustPilot, and SiteJabber. The pages of online shops were opened manually to check the availability of the favicon and the relationship with the online shop.

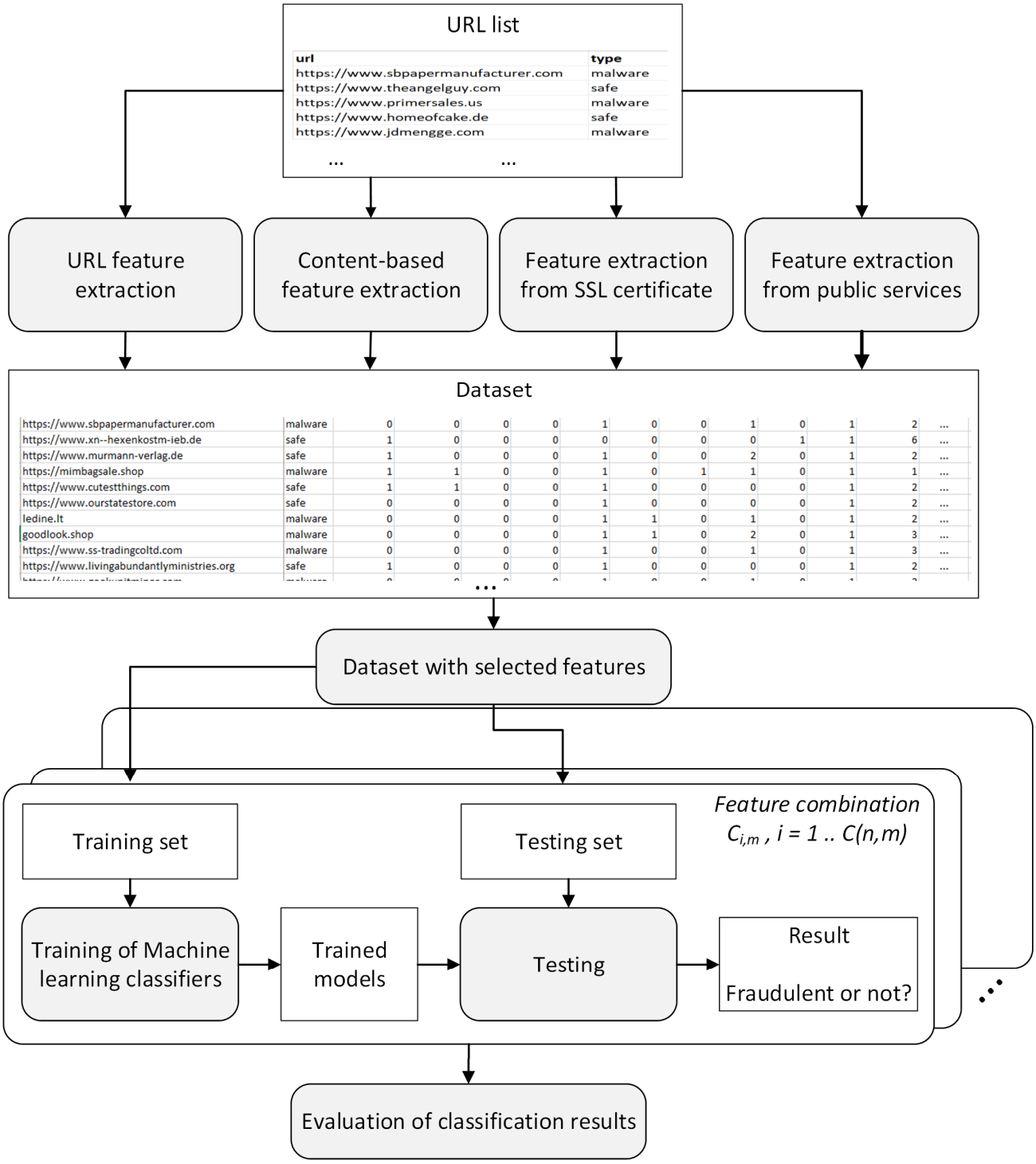

Figure 1. The methodology for evaluating possible combinations of features for the detection of fraudulent online shops.

In this study, seven URL-based features were used (Table 1). The URL-based features were obtained from the protocol and host domain, which consists of the second level

domain and the top-level domain. URL components such as the path and the query were eliminated and not included as features. The URLs of the internal pages of an online shop, such as the product description and shopping cart, usually consist of many different characters, but this does not indicate whether the shop is fraudulent or legitimate. The *F1* and *F2* features are related to the URL length. *F1* is the number of characters in the host domain name and *F2* is the number of characters in the top domain name. Legitimate online shops seek to obtain a short and memorable hostname. *F3* indicates whether the World Wide Web prefix is used in the URLs of the online shop. Legitimate online shops quite often use the URL without the prefix "www", and if the user enters the link with the prefix "www", the link is automatically shortened. The *F4–F7* features indicate how many different characters the URL contains, such as numbers, letters, dots, and hyphens. Characters, except numbers, letters, dots, and hyphens, are not allowed in domain names. *F8–F13* are content-based features. Four features are related to the payment methods on the website. F8 indicates whether the website offers payment using credit cards. *F9* indicates whether online shop buyers can use a payment method that allows them to get money back, including PayPal, Alipay, Apple Pay, Google Pay, Samsung Pay, and Amazon Pay. *F10* indicates whether the website offers payment methods that allow the buyer to pay for the goods when they are received. *F11* indicates whether the website allows payments using crypto currency. Legitimate online shops usually provide payment methods that encourage buyers to buy goods knowing that they can get their money back in the case of damaged, lost or unsuitable goods. In contrast, fake online shops just want to receive money from victims and offer anonymous payment methods that do not allow identifying the owner of the shop and requesting a return of the money for non-delivered goods. *F12* indicates that the website uses free email services, including Gmail, Hotmail, Outlook, Yahoo Mail, Zoho Mail, ProtonMail, iCloud Mail, GMX Mail, AOL Mail, mail.com, Yandex Mail, Mail2World, or Tutanota. Legitimate online shops are expected to use email addresses with the same domain as used in the online shop URL. *F13* shows whether the website uses its own favicon. Online shops use their logo as a favicon to be remembered by new customers. This URL icon is associated with the website, and customers can more easily recognize what the website is when browsing, which is very important for legitimate online shops.

The feature *F14* is related to the SSL certificate. *F14* indicates the organization of the issuer of the SSL certificate. In our prepared dataset, websites use SSL certificates issued by the 10 most common organizations. Therefore, the value of the *F14* feature is a number from 1 to 10 representing the ID assigned to the organization, while *ID = 11* is used for other not so common issuers.

The *F15* feature shows whether the domain is young and was registered 400 days ago or later based on WHOIS data. If the shop works for a long time with the same domain, there is a high possibility that the shop is legitimate, since it seeks to preserve a domain name known to its customers. Some legitimate online shops registered their domain name more than 25 years ago. While fraudulent online shops tend to set up quickly and take down their domain name once they have achieved their goals. However, due to data privacy, some online shops hide their domain registration data, so this feature uses three states: young, old, and unavailable.

The *F16* and *F17* features indicate whether the website has any reviews on the publicly available TrustPilot [31] and SiteJabber [35] review and rating platforms. These features indicate only the presence of reviews or a rating score. These features do not show whether the reviews are positive or whether the rating score is low or high. Users of legitimate online shops tend to leave reviews and comments, as short-lived illegal stores do not have time to get feedback from users. Although a legitimate online shop may have a low rating score, this does not necessarily involve fraudulent activity. To take the rating score into account, it is necessary to analyze the content of user reviews, which is not the aim of this work. The *F18* feature indicates whether the domain of the website is included in the standard Tranco list [34], which consists of one million domains based on the average number of visits in the last 30 days. The list includes domains with the highest number of

user requests and is updated monthly. Legitimate websites are constantly present in this list because they have a high volume of visitors, which is difficult for fraudulent online shops to achieve. Fraudulent online shops are not as well-known as legitimate because their activity period is too short. Therefore, the number of visitors to these shops is too low to include them in the Tranco list.

Table 1. Description of the features used in this study.

Features	Description	Possible Values
F1—Domain length	Number of symbols in the host domain name.	Number, [7 . . . 38] *
F2—Top domain length	Number of symbols in the top domain name.	Number, [2 . . . 13] *
F3—Presence of prefix "www"	Presence of the prefix 'www' in the active URL of the online shop.	{0, 1}
F4—Number of digits	Number of digits in the URL.	Number, [0 . . . 4] *
F5—Number of letters	Number of letters in the URL.	Number, [11 . . . 39] *
F6—Number of dots (.)	Number of dots in the URL.	Number, [1 . . . 3] *
F7—Number of hyphens (-)	Number of hyphens in the URL.	Number, [0 . . . 4] *
F8—Presence of credit card payment	Presence of payment methods, which offer the consumer the option to pay using credit cards.	{0, 1}
F9—Presence of money back payment	Presence of payment methods, which offer the consumer the option of getting their money back.	{0, 1}
F10—Presence of cash on delivery payment	Presence of payment methods, which allow the consumer to pay for goods once they are received.	{0, 1}
F11—Presence of crypto currency	Presence of the ability to use cryptocurrencies for payments.	{0, 1}
F12—Presence of free contact emails	Indication of whether public e-mail services are used for contact e-mail.	{0, 1, 2, 3} 0—email address not found 1—free email address 2—domain email address 3—other email address
F13—Presence of logo URL	Indication of whether the website uses its own favicon, which is associated with the online shop logo and is shown in the browser's address bar.	{0, 1}
F14—SSL certificate issuer organization	The ID of the organization of the SSL certificate issuer: 1—Cloudflare, Inc., 2—Let's Encrypt, 3—Sectigo Limited, 4—cPanel, Inc., 5—GoDaddy.com, Inc., 6—Amazon, 7—DigiCert, Inc., 8—GlobalSign nv-sa, 9—Google Trust Services LLC, 10—ZeroSSL, 11—other organization.	[1 . . . 11]
F15—Indication of young domain	Shows whether the domain is young, registered 400 days ago or later. Due to data protection, not all domain owners provide a date of registration; such domains are identified using a special value 'hidden'. The domain registration date comes from the WHOIS database.	{0, 1, 2} 0—'old' domain name 1—'young' domain name 2—'hidden'
F16—Presence of TrustPilot reviews	Indicates whether the website has at least one review on the TrustPilot platform.	{0, 1}
F17—Presence of SiteJabber reviews	Indicates whether the website has at least one review on the SiteJabber platform.	{0, 1}
F18—Presence in the standard Tranco list	Indicates whether the domain of the website is included in the standard Tranco list based on the average number of visits.	{0, 1}

* Value ranges in the used dataset.

2.3. Experimental Setup

To create the primary dataset, a Python script was written to extract the features based on URL, website content, SSL certificate, domain registration date, and public ranking and reviews. Python standard libraries (version 3.9.13) were used: 're', 'urllib', 'ssl', 'socket', and 'requests'. Furthermore, the libraries 'whois' (version 0.7.2), 'tld' (version 0.13), and 'Tranco' (version 0.6) were used, which are not included in the standard Python distribution. Finally, 'BeautifulSoup' (version 4.11.1) was used to scrape the source code of online shops, because this library has good documentation and code examples. On the other hand, other existing scraping libraries such as 'lxml', 'parsel' or 'Scrapy' could also be used, since only basic text parsing functionality is required to extract domain name, email, and payment methods. To create secondary datasets with different combinations of features, we used a 'combinations' function from 'itertools' library. This function constructs m-length tuples with no repeated elements. The secondary datasets were divided into training and testing sets, applying different splitting strategies as described in Section 2.1. These datasets were used to train and evaluate the machine learning classifiers from the Scikit-learn library (version 1.0.2) [48], namely Decision Tree, Random Forest, Stochastic Gradient Descent, Logistic Regression, Gaussian Naïve Bayes, and Multilayer Perceptron. According to [49], the Scikit-learn library is the most popular Python library for machine learning, which has the largest number of algorithms implemented in most categories. In addition, this library has extensive documentation and many source code samples available for reuse, therefore it was used as the main machine learning framework. The XGBoost library (version 1.7.3) [50] was used for the implementation of the Extreme Gradient Boosting classifier. The configuration parameters for each classifier are given in Table 2.

Table 2. The parameters of used classifiers.

Classifier	Parameters
DecisionTreeClassifier	(criterion = 'gini', splitter = 'best', max_depth = None, min_samples_split = 2, min_samples_leaf = 1, min_weight_fraction_leaf = 0.0, max_features = None, random_state = None, max_leaf_nodes = None, min_impurity_decrease = 0.0, class_weight = None, ccp_alpha = 0.0)
RandomForestClassifier	(n_estimators = 100, criterion = 'gini', max_depth = None, min_samples_split = 2, min_samples_leaf = 1, min_weight_fraction_leaf = 0.0, max_features = 'sqrt', max_leaf_nodes = None, min_impurity_decrease = 0.0, bootstrap = True, oob_score = False, n_jobs = None, random_state = None, verbose = 0, warm_start = False, class_weight = None, ccp_alpha = 0.0, max_samples = None)
SGDClassifier	(loss = 'hinge', penalty = 'l2', alpha = 0.0001, l1_ratio = 0.15, fit_intercept = True, max_iter = 1000, tol = 0.001, shuffle = True, verbose = 0, epsilon = 0.1, n_jobs = None, random_state = None, learning_rate = 'optimal', eta0 = 0.0, power_t = 0.5, early_stopping = False, validation_fraction = 0.1, n_iter_no_change = 5, class_weight = None, warm_start = False, average = False)
LogisticRegression	(penalty = 'l2', dual = False, tol = 0.0001, C = 1.0, fit_intercept = True, intercept_scaling = 1, class_weight = None, random_state = None, solver = 'liblinear', max_iter = 2000, multi_class = 'ovr', verbose = 0, warm_start = False, n_jobs = 1, l1_ratio = None)
GaussianNB	(priors = None, var_smoothing = 1×10^{-9})
MLPClassifier	(hidden_layer_sizes = (100,), activation = 'relu', solver = 'sgd', alpha = 0.0001, batch_size = 'auto', learning_rate = 'constant', learning_rate_init = 0.001, power_t = 0.5, max_iter = 2000, shuffle = True, random_state = None, tol = 0.0001, verbose = False, warm_start = False, momentum = 0.9, nesterovs_momentum = True, early_stopping = False, validation_fraction = 0.1, beta_1 = 0.9, beta_2 = 0.999, epsilon = 1×10^{-8}, n_iter_no_change = 10, max_fun = 15,000)
XGBoost	(base_score = 0.5, booster = 'gbtree', device = 'cpu', colsample_bylevel = 1, colsample_bynode = 1, colsample_bytree = 1, gamma = 0, interaction_constraints = '', learning_rate = 0.1, max_delta_step = 0, max_depth = 6, min_child_weight = 1, monotone_constraints = '()', n_estimators = 10, num_parallel_tree = 1, objective = 'binary:logistic', random_state = 0, reg_alpha = 0, reg_lambda = 1, scale_pos_weight = 1, subsample = 1, sampling_method = 'uniform', tree_method = 'auto', scale_pos_weight = 1, grow_policy = 'depthwise', max_leaves = 0, max_bin = 256, validate_parameters = 1, verbosity = 1, use_rmm = False)

The Anaconda (version 2.3.2) [51] distribution of the Python language together with Spyder (version 5.2.2) [52], the Scientific Python Development Environment, which is a free integrated development environment (IDE) included with Anaconda, was used. Development and computations were performed on the 64-bit Windows 10 machine with 16 GB RAM and an Intel Core i7-7700 CPU 3.60 GHz processor.

3. Results

The results of the experimental evaluation are presented in Table 3. The table shows the combinations of features that guaranteed the best result of the classification algorithms when the dataset consists of m features. For each number of features m, the highest classification accuracy was found and all combinations of features that allowed this accuracy are presented in Table 3. Therefore, the number of combinations presented in the table for each value of m in the table may differ. For example, there are six best combinations with 6 features, and only one best combination with 8 features. The 'X' represents the features that are included in the combination. For example, for the size of the feature set $m = 2$, the best result was shown by the sets {*F14*, *F15*} and {*F13*, *F15*} with a classification accuracy of 0.9211 for the decision tree and the random forest classifiers.

As we can see from the results presented in Table 3, the best classification accuracy of 0.9693 was provided by the sets of features where $m = 13$ and $m = 15$. In most cases, the best results were demonstrated by the random forest classifier, followed by decision tree and XGBoost. Although the best results were achieved using a large number of features, combinations with a smaller number of features showed only a few percent lower classification accuracy. As presented in Figure 2, starting from $m = 7$, the accuracy reaches 0.9605, and further increasing the number of features adds only 0.0088.

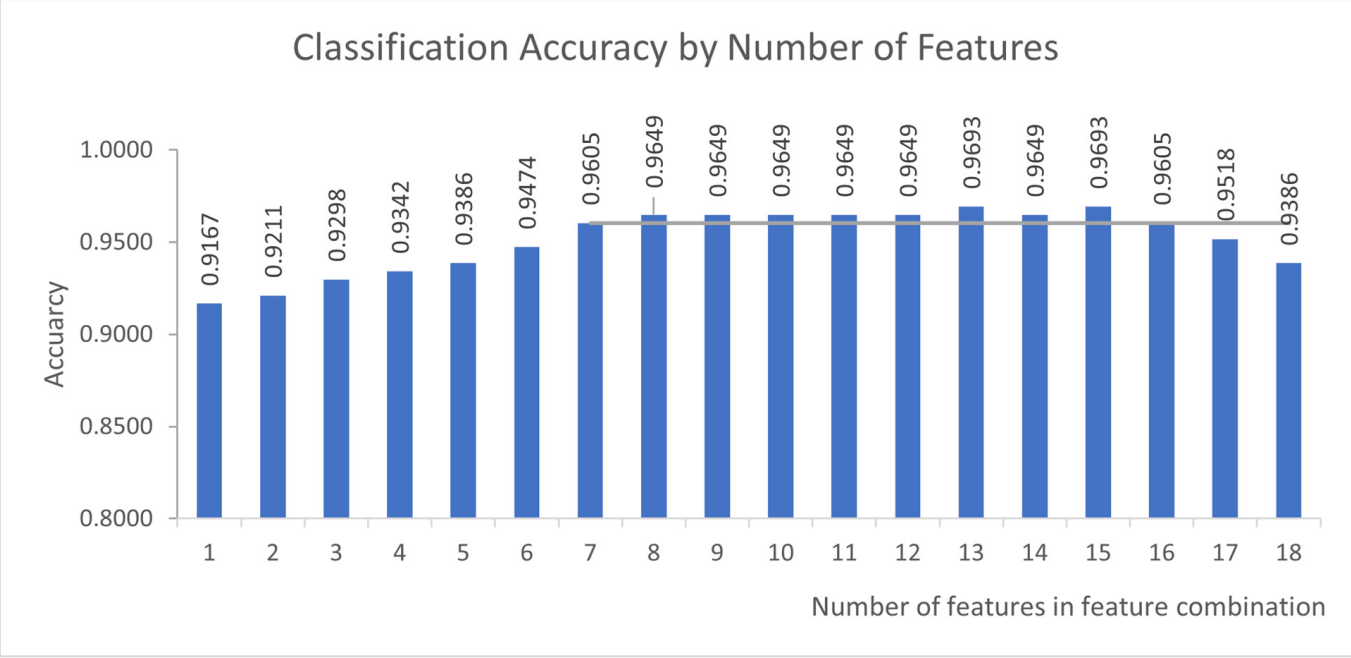

Figure 2. Best classification accuracy for different combinations of features, depending on the size of the combination.

The number of 'X' in the feature columns of Table 3 essentially shows how important this feature is in various combinations of features and leads to high classification accuracy. The significance of the features is presented in Figure 3, where the features are arranged according to the number of best combinations from Table 3 in which they have been used.

Table 3. The combinations of features that guarantee the highest classification accuracy. 'X' indicates in which combinations features are used. The highest accuracy values are presented in bold.

Number of Features	Features * Used in Feature Combinations																		Accuracy of Classifiers **						
	F1	*F2*	*F3*	*F4*	*F5*	*F6*	*F7*	*F8*	*F9*	*F10*	*F11*	*F12*	*F13*	*F14*	*F15*	*F16*	*F17*	*F18*	DT	RF	SGD	LR	GNB	MP	XGB
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	**0.9167**	**0.9167**	0.8596	0.8596	0.8596	0.8596	0.9079
2	-	-	-	-	-	-	-	-	-	-	-	-	X	-	X	-	-	-	**0.9211**	**0.9211**	0.8640	0.8640	0.6184	0.8640	0.9167
	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	-	-	-	**0.9211**	**0.9211**	0.8596	0.8509	0.8509	0.8816	0.8947
3	-	-	-	X	-	-	-	-	-	-	-	-	-	X	X	-	-	-	0.9211	**0.9298**	0.5307	0.8553	0.8596	0.8816	0.8947
	-	-	-	-	-	-	X	-	X	-	-	-	-	-	X	-	-	-	**0.9298**	**0.9298**	0.8596	0.8596	0.8596	0.8596	0.9254
4	-	-	-	X	-	-	X	-	X	-	-	-	-	-	X	-	-	-	**0.9342**	**0.9342**	0.8596	0.8596	0.8640	0.8772	0.9298
	-	-	-	-	-	-	X	-	X	-	-	-	X	-	X	-	-	-	**0.9342**	0.9254	0.6140	0.8509	0.6667	0.8772	0.9211
	-	-	-	-	-	-	X	-	X	-	-	-	-	-	X	-	X	-	0.9298	**0.9342**	0.8596	0.8596	0.5658	0.8728	0.9254
	-	-	-	X	-	-	X	-	-	-	-	-	-	X	X	-	-	-	0.9211	**0.9342**	0.7851	0.8553	0.8421	0.8816	0.8991
5	-	-	-	X	-	-	X	-	X	-	X	-	-	-	X	-	-	-	**0.9386**	0.9342	0.8596	0.8596	0.8596	0.8728	0.9298
	-	X	-	-	-	-	X	-	X	-	-	-	-	-	X	X	-	-	0.9342	**0.9386**	0.7895	0.8860	0.8333	0.9079	**0.9386**
	-	X	X	-	-	-	-	-	X	-	-	-	-	-	X	X	-	-	0.9342	**0.9386**	0.8640	0.8904	0.8596	0.8991	0.9298
	-	X	-	-	X	-	-	-	X	-	-	-	-	-	X	X	-	-	0.9167	0.9342	0.8553	0.8860	0.8684	0.9167	**0.9386**
	-	-	-	-	X	X	-	-	-	-	-	X	X	-	X	-	-	-	0.8816	0.9035	0.8421	0.8333	0.8816	0.8070	**0.9386**
6	-	X	-	-	X	X	-	-	X	-	-	-	-	-	X	X	-	-	0.9298	**0.9474**	0.5088	0.8728	0.8640	0.8991	0.9342
	-	X	X	-	-	-	X	-	X	-	-	-	-	-	X	X	-	-	0.9430	**0.9474**	0.8333	0.8904	0.8640	0.9079	0.9342
	-	X	-	-	X	X	-	-	X	-	-	-	X	-	X	-	-	-	0.9342	**0.9474**	0.7939	0.8377	0.8991	0.8904	0.9211
	-	X	-	-	-	-	X	-	X	-	-	-	X	-	X	X	-	-	0.9342	**0.9474**	0.8465	0.8860	0.8991	0.8947	0.9342
	-	X	X	-	-	-	-	-	X	-	-	-	X	-	X	X	-	-	0.9386	**0.9474**	0.7412	0.8860	0.9123	0.8947	0.9342
	-	X	-	-	-		-	-	X	X	-	-	X	-	X	X	-	-	0.9342	**0.9474**	0.8640	0.8816	0.8816	0.9123	0.9342
7	-	X	-	-	-	X	-	-	X	-	-	-	X	X	X	X	-	-	0.9254	**0.9605**	0.8684	0.8728	0.9079	0.8860	0.9254
	-	X	-	-	-	X	X	-	X	-	-	-	X	-	X	X	-	-	0.9342	**0.9605**	0.9035	0.8772	0.9167	0.8947	0.9386
8	-	X	X	-	-	-	X	-	X	-	-	-	X	X	X	X	-	-	0.9254	**0.9649**	0.8816	0.8728	0.9079	0.8860	0.9211

Table 3. *Cont.*

Number of Features	Features * Used in Feature Combinations																		Accuracy of Classifiers **					
	F1	*F2*	*F3*	*F4*	*F5*	*F6*	*F7*	*F8*	*F9*	*F10*	*F11*	*F12*	*F13*	*F14*	*F15*	*F16*	*F17*	*F18*	DT	RF	SGD	LR	GNB	MP	XGB
9	-	X	X	X	-	-	X	-	X	-	X	-	X	-	X	X	-	-	0.9298	**0.9649**	0.8947	0.8772	0.9211	0.8947	0.9386
	-	X	-	-	-	X	X	-	X	-	-	-	X	X	X	X	X	-	0.9211	**0.9649**	0.8904	0.8772	0.8553	0.8904	0.9342
	-	X	X	X	-	-	X	-	X	X	-	-	X	-	X	X	-	-	0.9254	**0.9649**	0.8640	0.8904	0.8947	0.9079	0.9342
10	X	X	-	-	-	X	-	X	X	-	-	X	X	-	X	X	X	-	0.9079	**0.9649**	0.8026	0.8904	0.8816	0.8904	0.9298
11	-	X	X	-	X	X	-	-	X	X	-	X	X	-	X	X	X	-	0.9254	**0.9649**	0.8947	0.9035	0.8684	0.8991	0.9298
	-	X	X	X	-	X	X	-	X	X	X	-	X	-	X	X	-	-	0.9342	**0.9649**	0.9298	0.8904	0.9035	0.9079	0.9386
12	-	X	X	-	X	X	X	X	X	-	X	-	X	X	X	X	-	-	0.9167	**0.9649**	0.5175	0.8947	0.8991	0.8728	0.9211
	-	X	X	-	X	X	-	X	X	-	X	-	X	X	X	X	X	-	0.9123	**0.9649**	0.8684	0.9035	0.8553	0.8772	0.9254
	-	X	X	X	-	-	X	X	X	-	X	-	X	X	X	X	X	-	0.9386	**0.9649**	0.9123	0.8904	0.8684	0.8772	0.9211
	X	X	-	X	-	X	-	X	X	X	-	X	X	-	X	X	X	-	0.9167	**0.9649**	0.8816	0.8947	0.8772	0.8991	0.9211
	-	X	X	X	X	X	-	X	X	-	X	-	X	X	X	X	-	-	0.9211	**0.9649**	0.8860	0.9035	0.9079	0.8772	0.9254
	-	X	-	-	X	X	-	X	X	X	X	X	X	-	X	X	X	-	0.9254	**0.9649**	0.8816	0.8991	0.8596	0.8991	0.9342
	-	X	X	X	-	-	X	X	X	X	X	-	X	X	X	X	-	-	0.9386	**0.9649**	0.8553	0.8816	0.8947	0.8860	0.9254
13	-	X	X	-	X	X	-	X	X	X	X	-	X	X	X	X	X	-	0.9211	**0.9693**	0.8596	0.8991	0.8596	0.8728	0.9167
14	X	X	X	X	-	X	-	X	X	X	-	X	X	-	X	X	X	X	0.9167	**0.9649**	0.8904	0.8991	0.8377	0.9079	0.9211
	-	X	X	X	-	X	X	X	X	X	X	-	X	X	X	X	-	X	0.9386	**0.9649**	0.8991	0.9079	0.8684	0.8816	0.9254
15	-	X	X	X	X	X	-	-	X	X	X	X	X	X	X	X	X	X	0.9167	**0.9693**	0.8947	0.8947	0.8333	0.9035	0.9123
16	-	X	X	X	X	X	X	X	X	X	X	X	X	-	X	X	X	X	0.9211	**0.9605**	0.8947	0.8991	0.8333	0.9079	0.9298
	X	X	X	X	X	-	X	X	X	X	X	X	X	-	X	X	X	X	0.9167	**0.9605**	0.8947	0.9035	0.8289	0.8947	0.9342
17	-	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0.9123	**0.9518**	0.8465	0.8947	0.8421	0.8947	0.9167
	X	X	X	X	-	X	X	X	X	X	X	X	X	X	X	X	X	X	0.9211	**0.9518**	0.8991	0.8947	0.8333	0.9035	0.9211
18	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0.8991	**0.9386**	0.9035	0.9167	0.8289	0.8947	0.9211

* Features: *F1—Domain length*, *F2—Top domain length*, *F3—Presence of prefix "www"*, *F4—Number of digits*, *F5—Number of letters*, *F6—Number of dots (.)*, *F7—Number of hyphens (-)*, *F8—Presence of credit card payment*, *F9—Presence of money back payment*, *F10—Presence of cash on delivery payment*, *F11—Presence of crypto currency*, *F12—Presence of free contact emails*, *F13—Presence of logo URL*, *F14—SSL certificate issuer organization*, *F15—Indication of young domain*, *F16—Presence of TrustPilot reviews*, *F17—Presence of SiteJabber reviews*, *F18—Presence in the standard Tranco list*; ** Classifiers: DT—Decision Tree, RF—Random Forest, SGD—Stochastic Gradient Descent, LR—Logistic Regression, GNB—Gaussian Naive Bayes, MP—Multilayer Perceptron, and XGB—XGBoost.

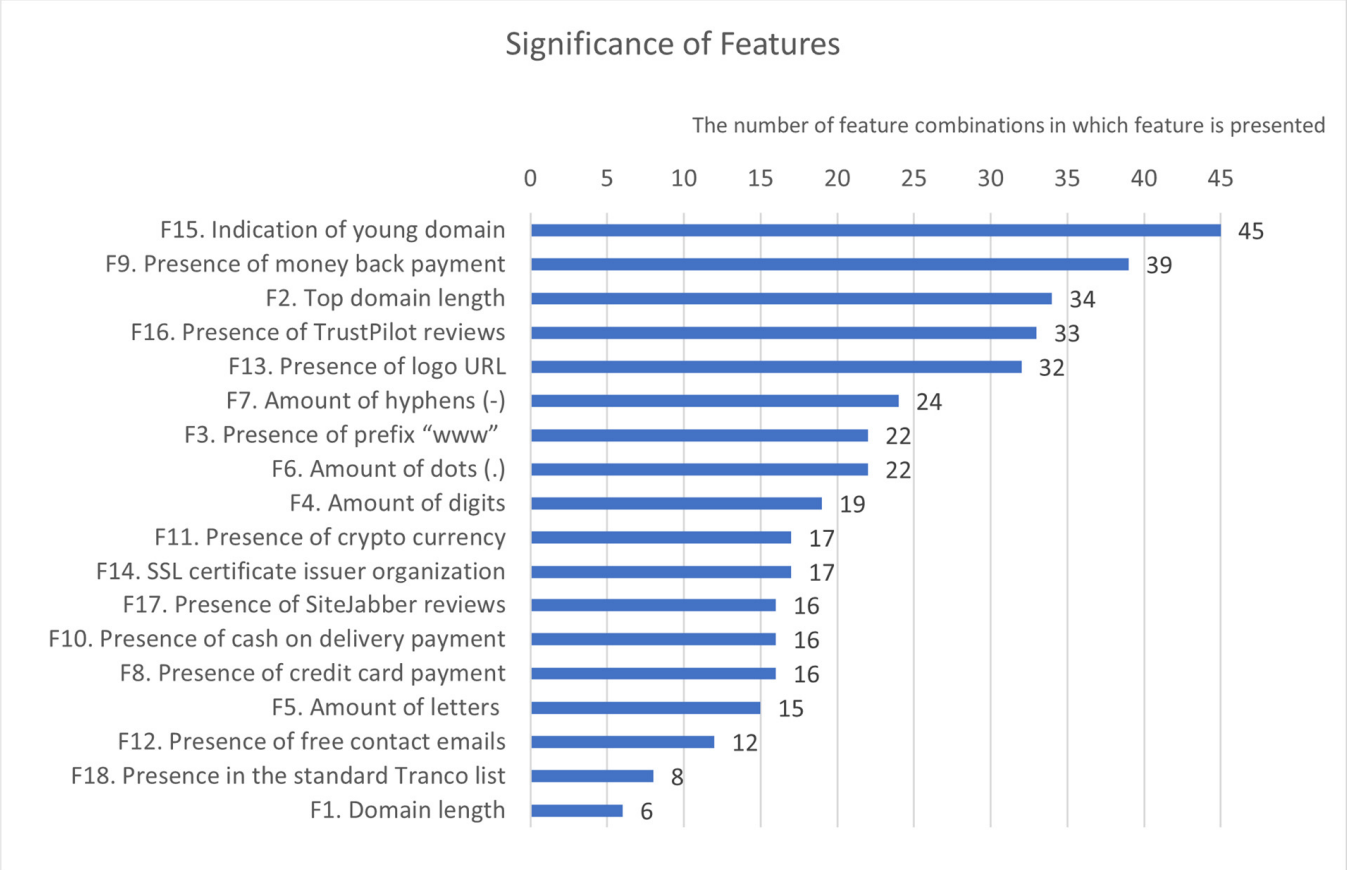

Figure 3. Significance of the features, based on the presence of the features in the combinations that showed the best classification results.

The best classification accuracies for the different number of features were achieved by 45 combinations. For example, all these combinations contain *F15—Indication of young domain* (the top feature in Figure 3), which is the most significant feature. The second most significant feature is *F9—Presence of money back payment*, which is included in 39 combinations of features. The third is *F2—Top domain length*, the fourth is *F16—Presence of TrustPilot reviews*, and so on. The five most significant features are presented in all the best feature combinations with a size greater than 6.

The results presented in this section allow us to conclude that even a small number of the most significant features is sufficient to achieve very high classification accuracy, avoiding complex computations and data analysis. All features can be obtained directly from the HTTP response data when accessing the online shop. Some features (*F1–F7*, *F13*) are extracted directly from the URL string, few features (*F14–F18*) are obtained by direct calls to third-party APIs, and some features (*F8–F12*) are extracted from the source code of the webpage simply by searching for keywords related to payment methods and emails. It means that the features used in this study are suitable for creating lightweight fraudulent online shop detection models, with a quick and simple feature extraction process. This is very important for practical applications, creating security apps and add-ons with memory and processing power limitations, e.g., Internet of Things, mobile applications, browser add-ons, etc. Developers of such solutions can choose to use only a few of the most significant features to make fast fraudulent-shop detection software with small memory footprint that still has very high accuracy. They can also use more features, increasing the size of the classification model and its accuracy, but still achieve a fast detection process.

4. Discussion

4.1. Context and Major Findings

In the beginning of this research, we raised the hypothesis that fraudulent shops can be detected using a relatively small number of features that can be easily obtained from third-party sources, external APIs, or fake shop source code directly. The idea was to avoid complex analysis and computations for feature extraction and keep the number of features as small as possible. This would allow us to apply this method in practical applications, where memory and processing capabilities are restricted, e.g., creating mobile security applications. We tried to find the smallest possible number of features, yet still achieving very high detection accuracy. For this purpose, we evaluated all possible combinations of features (from the initial set of 18 features), using seven popular classifiers. The initial set of 18 features was constructed in such a way that all features can be directly extracted from the URL and the source code of the online shop and the APIs of third parties using simple text analysis techniques.

As presented in Figure 2, even a small number of proposed features was enough to achieve high classification accuracy. The accuracy of 0.9342 was achieved with only four of the most significant features. Seven of the most significant features ensured a classification accuracy of 0.9605, while the best accuracy of 0.9693 was achieved using 13 and 15 features. Therefore, our research allowed us to identify the small set of the most significant features, which allows one to detect fake shops with high accuracy, avoiding complex, time-consuming computations and analysis of web shop content, such as, for example, comparing shop prices with other known legitimate e-shops [38].

4.2. Comparison to Similar Studies

The fraud and scam detection problem has been extensively studied in the scientific literature for many years. One of the types of such fraud is web phishing, where fraudsters try to steal user data or money by creating fake websites. Machine learning algorithms and methods are usually used to identify phishing websites together with publicly available datasets such as PhishTank [8] and VirusTotal [30]. Most of these studies do not distinguish online shops as a separate object of study and are more focused on methods for detecting phishing pages. In contrast to our study, these works use different features that focus more on sophisticated URL analysis [24] than website content analysis. Therefore, these phishing detection methods are not suitable for detecting fake online shops, because the latter requires evaluation of their content data. For the same reason, existing datasets are not suitable for training fraudulent online shop classifiers, so we had to create our own dataset for this study.

To our knowledge, there are few scientific works that are dedicated solely to identifying fake online shops. Unlike our proposed solution, [32,37] use only features extracted from the content and a complex analysis of the entire website code and its structure. The authors of [17] also perform textual and image analysis of online shops. Contrary to our study, this analysis is not a simple keyword search, but is based on natural language processing (NLP) methods. Although high accuracies (0.998, 0.97, and 0.987, respectively) are achieved in all these studies, these methods can be difficult to apply in practice due to the high processing and memory requirements. Therefore, in our study, we propose using features that can be obtained directly by querying third-party services (e.g., WHOIS [33]), analyzing the web address string (e.g., number of digits), or simply searching the source code of the webpage for specific parts and keywords (e.g., contact email, payment information).

The work most similar to our study is Sánchez-Paniagua et al. [38]. The authors of [38] also proposed to use features obtained from the source of the website and third parties, including Trustpilot [31] and WHOIS [33], but in addition they use data from social network platforms together with features obtained from the metadata of the website. These features include high discounts, social media footprint, domain age, registration date, SSL names, country and issuer, Trustpilot score and review, e-commerce technologies and policies. Contrary to our proposal, this study does not include URL-based features.

The method proposed in [38] achieved 0.86 accuracy using a custom made dataset containing 282 records.

4.3. Importance of the Features

This study confirmed some well-known facts regarding the fake e-shops features, but also allowed us to identify some less obvious dependencies between e-shop-related data and its trustworthiness. Usually, various authorities warn users that a very young website together with unusually small prices almost certainly means fraud. The results of this study completely confirm that the age of the online shop (*F15—Indication of young domain*) is the most significant feature, and even using this feature alone allows 0.9167 classification accuracy to be achieved. This means that the old domain of the shop in most cases indicates that this shop is trustworthy. However, the young domain age does not always mean that this is a fake shop, since some new legitimate shops appear in time. Therefore, we need additional features to distinguish young fake shops from young legitimate ones.

As mentioned before, we did not perform analysis of the goods prices, because it is complex, time-consuming, and very subjective, since even legitimate shops usually have high discounts, e.g., on Black Friday events. On the other hand, we tried to use various payment options as features, namely *F8—Presence of credit card payment*, *F9—Presence of money back payment*, and *F10—Presence of cash on delivery payment*. These features can be easily obtained by a simple analysis of the website of the shop, and as Figure 3 shows, the *F9—Presence of money back payment* is the second most significant feature. Simply speaking, only legitimate shops allow refunds, whereas fraudulent shops are only interested in quick money collection and theft. Credit card payment or cash on delivery do not definitely show if the shop is fraudulent, because attackers usually also have credit card payment in order to extract naive user data, while cash on delivery is not a very popular method even among legitimate shops.

Figure 3 also shows quite interesting results with respect to URL-based features. The *F2—Top domain length* feature is the third in importance and is presented in almost all feature combinations larger than four. We explain this by the fact that legitimate online shops use memorable domain names that are short and simple. Many of such domains have already been registered and used. Therefore, the creators of temporary fraudulent shops are forced to use longer domain names or to use names sounding similar to legitimate shops, usually using hyphens or dots in their names. As a result, the features *F7—Number of hyphens (-)* and *F6—Number of dots (.)* are also significant and take the sixth and eighth places in our ranking.

The presence (or absence) of user reviews in popular review and rating platforms seems to also play an important role. The *F16—Presence of TrustPilot reviews* feature is presented in almost all the best feature combinations, where the number of features is larger than 4, and is the fourth most significant feature in our ranking. Please note that we did not analyze the reviews and did not consider the reviews score, since even legitimate shops have many low-level reviews, for example due to delivery delays or wrong goods. This feature only shows if any reviews exist, since temporary fake shops usually have no time to accumulate such opinions.

The results also showed that the absence of the shop logo in the URL can indicate fraudulent online shops, as attackers do not waste their time doing such things, while many serious online shops put their logo in the browser's URL bar. Therefore, *F13—Presence of logo URL* is the fifth most important feature according to our results.

With larger feature combinations, containing more than eight features, the importance of the features is not so obvious since these sets contain features of all possible types. However, the results presented in Table 3 and Figure 2 allow us to draw the conclusion that a larger number of features does not lead to much higher classification accuracy, which is already achieved with the most significant features mentioned above.

The analysis of the less significant features in our dataset [36] shows that they do not unequivocally define fraudulent or legitimate online shops. These features have a lesser

impact on accuracy because they can be found in both legitimate and fraudulent shops' records of the dataset. For example, *F12—Presence of free contact emails* is not specific to fraudulent online shops, because there are a number of legitimate shops using free emails instead of emails based on their domain name. The same applies to the features *F1, F5, F8, F10,* and *F14.*

4.4. Comparison of Classifiers

Regarding the classification algorithms used for the experimental evaluation, we can see that all of the Logically Learning Algorithms are the best performers. The random forest classifier ('RF' in Table 3) gave the best results in all cases except four. It is followed by the decision tree ('DT' in Table 3), which was the best in seven cases and XGboost, which showed the best result three times. This can be explained by the structure of these classifiers that are tree-like and suit very well our selected features that are not closely interrelated. Moreover, the experimental results show that some independent features (such as *F15* and *F9*) are very important and it is very easy for these algorithms to make logic-based classification decisions.

On the other hand, the results of the statistics-based methods, such as Logistics Regression or Gaussian Naive Bayes, as well as ANN-based methods are showing both, the inferior overall accuracy as well as the lack of stability while using some specific combinations of features. Overall accuracy achieved by these methods rarely surpasses the mark of 0.9, but in some cases it can drop under 0.5. These results may be caused by the nature of the algorithms, which tries to find complex relations between features and the classification results but misses simple relations which are present in the data set.

4.5. Practical Applicability of the Results

Furthermore, the results presented in Table 3 suggest that we can implement the fake-shop detection algorithm even without a machine learning approach. Since the most significant features are presented almost in all the best combinations of features, the detection algorithm can be based on them and sound like a simple few-step instructions for the end user (or for the straightforward 'if-then-else' like implementation in the source code). For example, 'a legitimate online shop should be older than X months, have a money back option, and at least a few reviews on popular review platforms'. Such a simple suggestion will protect users from fraudulent online shops in most cases (more than 93%) even without a sophisticated analysis of the content of the online shop. Also, the decision can be based purely on public and freely available data, which can be quickly obtained even by not very experienced online users.

5. Conclusions

In this study, we have investigated the possibility of detecting fraudulent online shops using the minimal set of publicly and easily obtainable features. The goal was to create high-precision classification solutions that require little computing and memory resources. We evaluated various combinations of 18 features that were used to create several machine learning-based classification models. The best combinations of features were found by comparing the accuracy of the classifiers, and the most significant features were identified. We have obtained encouraging results demonstrating that even only four of the most significant features allow us to achieve 0.9342 classification accuracy, while 0.9605 accuracy is reached only with seven features. Our results could be applied to create lightweight fraudulent online shop detection models, with a quick and simple feature extraction process. This is very important for practical applications, creating security apps and add-ons with memory and processing power limitations, e.g., mobile security applications. The future research direction could be related to additional data collection, expanding our custom dataset, which was created for this study. Our results are promising and should be validated by a larger sample size, including data taken from a wider variety of regions.

Author Contributions: Conceptualization, A.L., G.D. and N.M.; formal analysis, N.M.; investigation, A.J. and G.D.; methodology, A.J., A.L. and N.M.; software, A.J. and G.D.; supervision, A.L.; validation, A.J., A.L., G.D. and N.M.; visualization, A.J.; writing—original draft, A.J. and A.L.; writing—review and editing, A.J., A.L. and N.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are openly available in Mendeley Data at https://www.doi.org/10.17632/m7xtkx7g5m.1 (accessed on 17 January 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hilal, W.; Gadsden, S.A.; Yawney, J. Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances. *Expert Syst. Appl.* **2022**, *193*, 116429. [CrossRef]
2. Al-Hashedi, K.G.; Magalingam, P. Financial Fraud Detection Applying Data Mining Techniques: A Comprehensive Review from 2009 to 2019. *Comput. Sci. Rev.* **2021**, *40*, 100402. [CrossRef]
3. Tang, L.; Mahmoud, Q.H. A Survey of Machine Learning-Based Solutions for Phishing Website Detection. *Make* **2021**, *3*, 672–694. [CrossRef]
4. Zieni, R.; Massari, L.; Calzarossa, M.C. Phishing or Not Phishing? A Survey on the Detection of Phishing Websites. *IEEE Access* **2023**, *11*, 18499–18519. [CrossRef]
5. Coppola, D. Global Number of Digital Buyers 2014–2021. Available online: https://www.statista.com/statistics/251666/number-of-digital-buyers-worldwide/ (accessed on 30 April 2023).
6. Coppola, D. Share of Online Shopping Scam Victims Who Lost Money Worldwide 2015–2022. Available online: https://www.statista.com/statistics/1273302/consumers-who-lost-money-due-to-online-shopping-scams/ (accessed on 30 April 2023).
7. Chevalier, S. Median Monetary Loss per Online Purchase Scam Worldwide 2015–2022. Available online: https://www.statista.com/statistics/1273330/median-money-lost-to-online-purchase-scams/ (accessed on 30 April 2023).
8. PhishTank. Available online: https://www.phishtank.com (accessed on 30 April 2023).
9. Alexa. Available online: https://www.alexa.com (accessed on 5 April 2023).
10. UCI Machine Learning Repository. Available online: https://archive.ics.uci.edu/ml/index.php (accessed on 30 April 2023).
11. OpenPhish. Available online: https://openphish.com/ (accessed on 30 April 2023).
12. Common Crawl Index Server. Available online: https://commoncrawl.org/ (accessed on 30 April 2023).
13. URL Dataset (ISCX-URL2016). Available online: https://www.unb.ca/cic/datasets/url-2016.html (accessed on 30 April 2023).
14. Ishikawa, T.; Liu, Y.-L.; Shepard, D.L.; Shin, K. Machine Learning for Tree Structures in Fake Site Detection. In Proceedings of the 15th International Conference on Availability, Reliability and Security, Virtual Event Ireland, 25 August 2020; ACM: New York, NY, USA; pp. 1–10.
15. Al-Sarem, M.; Saeed, F.; Al-Mekhlafi, Z.G.; Mohammed, B.A.; Al-Hadhrami, T.; Alshammari, M.T.; Alreshidi, A.; Alshammari, T.S. An Optimized Stacking Ensemble Model for Phishing Websites Detection. *Electronics* **2021**, *10*, 1285. [CrossRef]
16. Tanaka, S.; Matsunaka, T.; Yamada, A.; Kubota, A. Phishing Site Detection Using Similarity of Website Structure. In Proceedings of the IEEE Conference on Dependable and Secure Computing (DSC), Aizuwakamatsu, Fukushima, Japan, 30 January 2021; pp. 1–8.
17. Khoo, E.; Zainal, A.; Ariffin, N.; Kassim, M.N.; Maarof, M.A.; Bakhtiari, M. Fraudulent E-Commerce Website Detection Model Using HTML, Text and Image Features. In Proceedings of the 11th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2019), Hyderabad, India, 13–15 December 2019; Abraham, A., Jabbar, M.A., Tiwari, S., Jesus, I.M.S., Eds.; Advances in Intelligent Systems and Computing. Springer International Publishing: Cham, Switzerland, 2021; Volume 1182, pp. 177–186, ISBN 978-3-030-49344-8.
18. Chen, J.-L.; Ma, Y.-W.; Huang, K.-L. Intelligent Visual Similarity-Based Phishing Websites Detection. *Symmetry* **2020**, *12*, 1681. [CrossRef]
19. Chiew, K.L.; Chang, E.H.; Sze, S.N.; Tiong, W.K. Utilisation of Website Logo for Phishing Detection. *Comput. Secur.* **2015**, *54*, 16–26. [CrossRef]
20. Mostard, W.; Zijlema, B.; Wiering, M. Combining Visual and Contextual Information for Fraudulent Online Store Classification. In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, Thessaloniki, Greece, 14 October 2019; pp. 84–90.
21. Rendall, K.; Nisioti, A.; Mylonas, A. Towards a Multi-Layered Phishing Detection. *Sensors* **2020**, *20*, 4540. [CrossRef] [PubMed]
22. Jain, A.K.; Gupta, B.B. Phishing Detection: Analysis of Visual Similarity Based Approaches. *Secur. Commun. Netw.* **2017**, *2017*, 5421046. [CrossRef]

23. Aljofey, A.; Jiang, Q.; Qu, Q.; Huang, M.; Niyigena, J.-P. An Effective Phishing Detection Model Based on Character Level Convolutional Neural Network from URL. *Electronics* **2020**, *9*, 1514. [CrossRef]
24. Butnaru, A.; Mylonas, A.; Pitropakis, N. Towards Lightweight URL-Based Phishing Detection. *Future Internet* **2021**, *13*, 154. [CrossRef]
25. Kumar, J.; Santhanavijayan, A.; Janet, B.; Rajendran, B.; Bindhumadhava, B.S. Phishing Website Classification and Detection Using Machine Learning. In Proceedings of the International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 22–24 January 2020; pp. 1–6.
26. Sahingoz, O.K.; Buber, E.; Demir, O.; Diri, B. Machine Learning Based Phishing Detection from URLs. *Expert Syst. Appl.* **2019**, *117*, 345–357. [CrossRef]
27. Yang, R.; Zheng, K.; Wu, B.; Wu, C.; Wang, X. Phishing Website Detection Based on Deep Convolutional Neural Network and Random Forest Ensemble Learning. *Sensors* **2021**, *21*, 8281. [CrossRef] [PubMed]
28. ScamAdviser. Available online: https://www.scamadviser.com/ (accessed on 12 May 2023).
29. URLVoid. Website Reputation Checker. Available online: https://www.urlvoid.com/ (accessed on 12 May 2023).
30. VirusTotal. Available online: https://www.virustotal.com (accessed on 12 May 2023).
31. Trustpilot. Available online: https://www.trustpilot.com (accessed on 12 May 2023).
32. Shin, K.; Ishikawa, T.; Liu, Y.-L.; Shepard, D.L. Learning DOM Trees of Web Pages by Subpath Kernel and Detecting Fake E-Commerce Sites. *Make* **2021**, *3*, 95–122. [CrossRef]
33. WHOIS. Available online: https://who.is/ (accessed on 5 December 2023).
34. Le Pochat, V.; Van Goethem, T.; Tajalizadehkhoob, S.; Korczynski, M.; Joosen, W. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In Proceedings of the Network and Distributed System Security Symposium, San Diego, CA, USA, 24–27 February 2019.
35. Sitejabber. Available online: https://www.sitejabber.com/ (accessed on 11 July 2023).
36. Janaviciute, A.; Liutkevicius, A. Fraudulent and Legitimate Online Shops Dataset. Mendeley Data, 2023, V1. [CrossRef]
37. Beltzung, L.; Lindley, A.; Dinica, O.; Hermann, N.; LindJner, R. Real-Time Detection of Fake-Shops through Machine Learning. In Proceedings of the IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10 December 2020; pp. 2254–2263.
38. Sánchez-Paniagua, M.; Fidalgo, E.; Alegre, E.; Jáñez-Martino, F. Fraudulent E-Commerce Websites Detection Through Machine Learning. In *Hybrid Artificial Intelligent Systems*; Sanjurjo González, H., Pastor López, I., García Bringas, P., Quintián, H., Corchado, E., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2021; Volume 12886, pp. 267–279. ISBN 978-3-030-86270-1.
39. Metz, C.E. Basic Principles of ROC Analysis. *Semin. Nucl. Med.* **1978**, *8*, 283–298. [CrossRef] [PubMed]
40. Watchlist Internet. Available online: https://www.watchlist-internet.at (accessed on 11 January 2024).
41. Artists Against 419. Fake Sites List. Available online: https://db.aa419.org (accessed on 15 January 2024).
42. Global E-Commerce Websites List. Available online: https://www.kaggle.com/datasets/wiredwith/websites-list (accessed on 15 January 2024).
43. Online Shopping with Trusted Shops. Available online: https://www.trustedshops.eu/ (accessed on 15 January 2024).
44. The Ecommerce Europe Trustmark. Available online: https://ecommercetrustmark.eu/ (accessed on 15 January 2024).
45. EHI Geprüfter Online-Shop. Available online: https://ehi-siegel.de/ (accessed on 15 January 2024).
46. Retail Excellence Ireland. Available online: https://www.retailexcellence.ie/ (accessed on 15 January 2024).
47. Similarweb. Available online: https://www.similarweb.com/ (accessed on 15 January 2024).
48. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
49. Stancin, I.; Jovic, A. An Overview and Comparison of Free Python Libraries for Data Mining and Big Data Analysis. In Proceedings of the 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 20–24 May 2019; pp. 977–982.
50. XGBoost Documentation. Available online: https://xgboost.readthedocs.io/en/latest/index.html (accessed on 15 January 2024).
51. Anaconda. Available online: https://www.anaconda.com/ (accessed on 12 May 2023).
52. Spyder. The Scientific Python Development Environment. Available online: https://www.spyder-ide.org/ (accessed on 12 May 2023).