Hybrid Rule-Based Model for Phishing URLs Detection

Kayode S. Adewole[1(✉)], Abimbola G. Akintola[1], Shakirat A. Salihu[1], Nasir Faruk[2], and Rasheed G. Jimoh[1]

[1] Department of Computer Science, University of Ilorin, Ilorin, Nigeria
{adewole.ks, akintola.ag, salihu.sa, jimoh_rasheed}@unilorin.edu.ng
[2] Department of Telecommunications Science, University of Ilorin, Ilorin, Nigeria
nasirfaruk@gmail.com

Abstract. Phishing attack has been considered as a major security challenge facing online community due to the different sophisticated strategies that is being deployed by attackers. One of the reasons for creating phishing website by attackers is to employ social engineering technique that steal sensitive information from legitimate users, such as the user's account details. Therefore, detecting phishing website has become an important task worthy of investigation. The most widely used blacklist-based approach has proven inefficient. Although, different models have been proposed in the literature by deploying a number of intelligent-based algorithms, however, considering hybrid intelligent approach based on rule induction for phishing website detection is still an open research issue. In this paper, a hybrid rule induction algorithm capable of separating phishing websites from genuine ones is proposed. The proposed hybrid algorithm leverages the strengths of both JRip and Projective Adaptive Resonance Theory (PART) algorithm to generate rule sets. Based on the experiments conducted on two publicly available datasets for phishing detection, the proposed algorithm demonstrates promising results achieving accuracy of 0.9453 and 0.9908 respectively on the two datasets. These results outperformed the results obtained with JRip and PART. Therefore, the rules generated from the hybrid algorithm are capable of identifying phishing links in real-time with reduction in false alarm.

Keywords: Phishing website · JRip · PART · Machine learning · Rule-based model · Rule induction

1 Introduction

The recent development in information technology coupled with the advancement in Internet usage has created an ample of opportunity for online community to communicate and share varieties of resources. There has been an exponential growth in the number of businesses and organization offering web services to improve their customers' experience. Many of these organizations provide online trading including sales of goods and services over the World Wide Web (WWW) [1]. To access these online

M. H. Miraz et al. (Eds.): iCETiC 2019, LNICST 285, pp. 119–135, 2019.
https://doi.org/10.1007/978-3-030-23943-5_9

resources, users need to know their Uniform Resource Locators (URLs). URL is an essential identification for all objects on the WWW such as audio, video, hypertext pages and a host of other online resources. Nevertheless, despite the huge opportunities offered by the Internet, accessibility to online resources may expose Internet users to different forms of vulnerabilities and online threats. This can damage financial reputation and lead to loss of private information through various malicious strategies that may be deployed by hackers. Such strategies include the creation of phishing websites to lure legitimate users. Thus, the suitability of the Internet as a channel for secured online communication and commercial exchange posed a serious question. According to Dhamija, Tygar [2], phishing is categorized as a form of online threat that involves an act of impersonating a website or web resources of a reputable organization with the aim of illegally obtaining user's confidential information like social security numbers, usernames, and passwords. Phishing links are sometimes referred to as malicious URLs.

Attackers make use of malicious links in high magnitude to distribute malware over the web and to hijack confidential information from Internet users. If successful, the link can give partial or full control of the system to the attacker [3]. In recent years, there has been an increase in the growth of cybercrime which needs to be critically addressed by network information security authorities. Attackers have targeted many sectors from e-commerce and banking to government, private and many more by inserting malicious codes into a standard webpage to evade detection [4]. Timely detection of such phishing URLs is of great importance in order to reduce the damage it can cause to online community [5].

Early detection approach for phishing website was based on blacklist method, which relies on repository of already classified websites. This approach suffers from inclusiveness due to the fact that any URL or new URL that is not listed in the repository might evade detection [5, 6]. Machine learning approaches have also been deployed to build intelligent models that can separate phishing websites from legitimate ones. For instance, Gupta [7] applied pattern matching algorithm based on word segmentation to identify malicious URL. Thakur, Meenakshi [8] developed a system for detecting malicious URLs in big data environment using JRip rule induction machine learning algorithm. The detection and classification of malicious URLs in cloud environment based on machine learning approach has been investigated by [9]. The authors proposed a method that is based on Markov decision process, Information gain ratio and Decision tree to simultaneously analyzed malicious webpages. Although, a number of studies have applied machine learning algorithms to develop suitable models for phishing URLs detection, however, investigating hybrid predictive model to effectively detect phishing websites still remain an open research issue. Thus, this paper proposes hybrid rule-induction algorithm to address this research area.

The remaining parts of this paper are organized as follows: Sect. 2 discusses related studies on phishing website detection; Sect. 3 focuses on the main methodology deployed to develop the proposed hybrid rule-induction algorithm. Section 4 presents the results obtained from the different experiments conducted in this study, and finally Sect. 5 concludes the paper and presents future direction.

2 Related Work

Rules induction technique is categorized into two, namely, direct and indirect techniques. The direct technique involves rules generation directly from the data while indirect technique deals with rule generation from another classification algorithm. Vijayarani and Divya [10] examined the performance of three rule-based algorithms for breast cancer and heart disease diagnosis. Formally, let k represent an observation from the dataset, then an instance k can be detected by a rule r provided all the conditions in r according to the value of the attribute pair can also be satisfied based on the corresponding value of the attribute for instance k. Let C be a concept (i.e. decision) which represents the consequent of rule r, then a rule set R is said to completely covered the concept C provided every instance k an element of C has a rule r from R that covers k. Furthermore, it can equally be said that a rule set R is complete provided R covers every concept in the dataset [11]. Generally, rule induction algorithms belong to two major classes: global and local. The global rule induction algorithms used the set of all attribute values as the search space, while the local rule induction algorithms used the set of attribute-value pairs to explore the search domain. Many rule induction algorithms have been introduced over the years, which include Learning from Examples Module, version 1 and 2 - LEM1 and LEM2.

This section discussed some of the existing research efforts for phishing URL detection. In the model developed by Lee and Kim [12], the researchers examined the malicious URL's in a twitter stream. The study focused on exploring frequently shared URLs to discover the suspiciousness of correlated URL redirect chains. The authors experimented with various tweets extracted from the twitter timeline and a classifier was built around them. Experimental result shows that their proposed classification method was able to accurately detect suspicious URLs in a tweet. Spam message and spam account detection models on Twitter have also been proposed in the literature [13, 14]. Another model from Bhardwaj, Sharma [15] applied Artificial Bee Colony to detect malicious URLs. The study was able to detect whether target website is genuine or not with the notion that once a user is aware of the safety of any link they want to click, then half of the problem is solved.

The use of lexical analysis has also been proposed in the literature for detecting malicious web pages. In a research carried out by Darling, Heileman [16], lexical analysis of URLs were used to classify malicious web pages. This approach is light weight with the aim of exploring the best classification accuracy of a purely lexical analysis that could be used in real-time. This approach is only based on lexical features. A study on detecting malicious URLs on two-dimensional barcodes was conducted by Xuan and Yongzhen [17]. Their model was based on utilizing a hash function. The system was able to detect malicious URLs by first extracting the eigenvalues of malicious and benign URLs. Using this approach, a black and white list library was built. Safety tips were incorporated to the system for users according to the match rules generated. Their experiment was able to detect malicious URLs in two-dimensional barcodes. In the study conducted by Dewan and Kumaraguru [18], Facebook Inspector is proposed to identify malicious posts on Facebook social network in real-time. Dataset containing over four million public posts in news making event generated on

Facebook were used. They figured out two set of malicious posts, the one that is based on URL blacklists and Human annotations. These posts were run through a two-fold filtering process and this is confirmed through a cross-validation process of the supervised learning models. Using the developed models, a Facebook inspector was built to detect malicious posts in real-time with accuracy of 80%. Abdelhamid, Ayesh [1] proposed a technique based on associative classification to detect phishing websites. In this study, a Multi-label Classifier based Associative Classification (MCAC) was developed to test its capability for phishing detection. MCAC outperformed other intelligent algorithms evaluated in this study. A number of features for phishing website detection has been investigated in the study conducted by [19]. A study carried out by Gupta [7] applied Boyer Moore string pattern matching technique for word segmentation. In this study, the nature of the attack is detected as a phishing link, follows by the use of real-time system to obtain the phishing links from the DNS server. Finally, the word segmentation approach is used to identify malicious URL. A two stage classification system for detecting malicious URLs has been proposed in the work of [5]. The first phase was conducted with the aim of estimating the maliciousness of web pages and then forward to the next phase to identify the malicious web pages.

Although several studies have investigated the possibility of detecting phishing websites with each study proposing specific individual learning algorithm. However, the investigation of hybrid methods for identifying phishing URLs still remains an open research issue. Therefore, this paper proposes a hybrid-rule induction algorithm that is based on the fusion of two widely used rule induction techniques: JRip and PART. The proposed hybrid model guarantee promising results based on the different experiment conducted.

3 Methodology

Rule induction belongs to machine learning domain where formal rules are induced from a set of data instances. These rules represent patterns in the data or a scientific model of the data. It is one of the most essential techniques in data mining and machine learning, which is useful in extracting hidden patterns and relationships in a dataset. The proposed hybrid rule-based model in this study combines rules induced by JRip and PART algorithms as shown in Fig. 1. From this figure, data collected from different servers such as Yahoo, Alexa, Common Crawl, PhishTank and OpenPhish are preprocessed in order to extract meaningful features that can be used for categorizing phishing websites from legitimate ones. Features extracted from the data are provided for rule induction using both JRip and PART algorithms. These rules are evaluated to ascertain their applicability for the classification task. Rules from the two algorithms are merged to produce hybrid rule-based model with strong capability to detect Phishing URLs. The subsequent section discussed the datasets used for evaluating the proposed hybrid rule-based model.

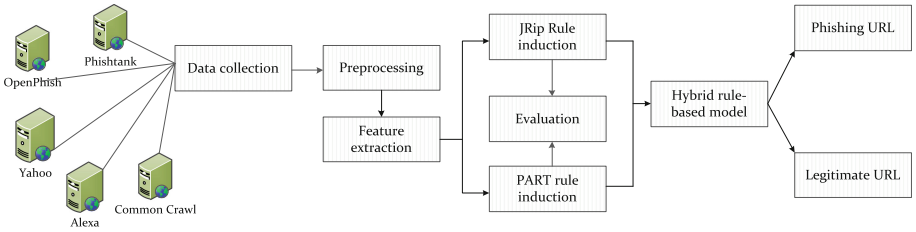

Fig. 1. Proposed hybrid rule-based model for phishing URLs detection.

3.1 Data Collection

This study analyzed two public datasets for phishing URLs detection in order to evaluate the performance of the proposed hybrid rule-based model. These datasets are available on the UCI repository. The first dataset, hereafter referred to as PhishingDataset1, is available at "https://archive.ics.uci.edu/ml/machine-learning-databases/00379/" which contains a total of 1353 URLs. Out of the 1353 URLs, 548 were identified as legitimate URLs as provided from Yahoo website while 702 and 103 URLs were identified as phishing and suspicious respectively from PhishTank. This dataset contains ten (10) features for analysis and was donated by [1]. The second dataset, hereafter referred to as PhishingDataset2, is available at "https://archive.ics.uci.edu/ml/machine-learning-databases/00327/". This dataset contains 4898 phishing URLs and 6157 legitimate URLs making a total of 11,055 URLs [19]. Table 1 shows the description of the two datasets considered in this research.

Table 1. Composition of the datasets used in this research.

Dataset name	No. of attributes	Attributes characteristics	No. of instances	Class distribution
PhishingDataset1	10	Integer	1,353	Phishing (702), legitimate (548), suspicious (103)
PhishingDataset2	30	Integer	11,055	Phishing (4898), legitimate (6157)

3.2 Phishing Website Features

To develop effective classification model, it is essential to ascertain the features that can guarantee the prediction of the class label with acceptable level of accuracy. This study utilized 10 and 30 features from PhishingDataset1 and PhishingDataset2 respectively. PhishingDataset2 contains all the features in PhishingDataset1 with addition of 20 features. Therefore, Table 2 shows the description of the features available in the two datasets. Asterisk (*) in the feature name indicates the features that are available in PhishingDataset1. This study uses the two datasets for developing the proposed hybrid rule-based model because they have similar features for analysis.

Table 2. Features in PhishingDataset1 and PhishingDataset2

Feature name	Feature category	Description
*having_IP_Address	Address bar	This is one of the address bar features whose presence in a URL may indicate phishing attack. For instance, http://128.87.2.100/crawl.html
*URL_Length	Address bar	If the length of a URL is long, this may indicate phishing attack. Average URL length of 54 is considered
Shortining_Service	Address bar	Address bar feature that indicates if a URL is shorten or not. Shorten URLs that link to long URL is considered phishing
having_At_Symbol	Address bar	Address bar feature whose presence in a URL indicates phishing attack since @ symbol can cause a web browser to ignore everything after @
double_slash_redirecting	Address bar	Address bar feature whose presence in a URL, excluding the one that follows HTTP, indicates phishing attack. For instance, http://www.normalurl.com//http://www.phishingweb.com
Prefix_Suffix	Address bar	Its presence in a URL indicates phishing attack. This is usually indicated with the use of dash (-)
having_Sub_Domain	Address bar	Multiple sub domains indicating phishing attack. This is usually indicated with the use of dot (.)
*SSLfinal_State	Address bar	URL without SSL indicated by HTTPS is considered phishing while those with HTTPS but with untrusted certificate issuer is considered suspicious
Domain_registeration_length	Address bar	If domain in the URL expires in less than a year, the URL is considered phishing
Favicon	Address bar	If the favicon displayed from the domain is at variant from that in the address bar, such URL is considered phishing
Port	Address bar	If the open port on the server is not within the preferred status, the URL is considered phishing.

(*continued*)

Table 2. (*continued*)

Feature name	Feature category	Description
HTTPS_token	Address bar	If HTTPS is added to the domain path, the URL is considered phishing. E.g. http://https-www.mypay-pay-creditcard.com
*Request_URL	Abnormal	A link is considered phishing if the percentage of Request URL is high. This deals with the number of external links embedded within the webpage
*URL_of_Anchor	Abnormal	The higher the number of URLs with anchor, the more suspicious the URL is
Links_in_tags	Abnormal	If the percentage links in tags such as <meta>, <script> and <link> is high, the URL is phishing
*SFH	Abnormal	If Server Form Handler (SFH) is empty, blank or refers to a dissimilar domain, the link is phishing or suspicious
Submitting_to_email	Abnormal	If the URL uses mail() or mailto: to submit user's data, it is considered phishing
Abnormal_URL	Abnormal	If the host name is not part of the URL, the link is considered phishing
Redirect	HTML/JavaScript	If the number of redirect of a URL is high such link is considered phishing
on_mouseover	HTML/JavaScript	If onMouseOver event causes the status bar to change, the URL is considered phishing
RightClick	HTML/JavaScript	Disabling right clicking is an indication of phishing
*popUpWidnow	HTML/JavaScript	The presence of popup window with text field is an indication of phishing
Iframe	HTML/JavaScript	The presence of Iframe is an indication of phishing
*age_of_domain	Domain	If age of a domain is less than 6 months, the URL is considered suspicious. This is extracted from WHOIS
DNSRecord	Domain	Absence of DNS record through the WHOIS query indicates phishing URL

(*continued*)

Table 2. (*continued*)

Feature name	Feature category	Description
*web_traffic	Domain	If the website is not ranked among the top 100,000 according to Alexa database rank, the URL is suspicious
Page_Rank	Domain	PageRank is a normalized value from 0 to 1 to measure the importance of a webpage. PageRank less than 0.2 is considered phishing
Google_Index	Domain	Webpage that is not indexed by Google Index is considered phishing due to the short life span
Links_pointing_to_page	Domain	If the number of links pointing to a webpage is less than 2, the URL is considered phishing
Statistical_report	Domain	If a URL is ranked among the top in the statistics from PhishTank or StopBadware, the URL is considered phishing
*Result		Feature indicating the class distribution. The value of 0 is suspicious, 1 is legitimate and −1 is phishing

3.3 Rule Induction Algorithms

As stated in the previous sections, this study considered two rule induction algorithms: JRip and PART due to their simplicity and performance as reported in the literature [20].

JRip

JRip is a rule induction algorithm introduced by William W. Cohen in [21]. JRip is an implementation of a propositional rule learner that is based on a Repeated Incremental Pruning to Produce Error Reduction (RIPPER). The algorithm provides an optimal version for the Incremental Reduced Error Pruning (IREP) algorithm. This rule induction algorithm directly extracts rules from the dataset based on propositional rule learning approach. The algorithm executes four main phases: growth, pruning, optimization and selection. The algorithm is described using the following pseudocode [20]:

Algorithm 1. JRip rule induction algorithm

```
Input: Pos (positive instances), Neg (negative instances)
Output: RS -> set of rules

Module BUILDRS (Pos,Neg)
Pos=positive instances
Neg=negative instances
RS= { }
DL_LENGTH=Desc_length (RS, Pos, Neg)
    DOWHILE Pos is not { }
    //New rule growing and pruning
    split (Pos,Neg) into (PosGrow, NegGrow) and (PosPrune, NegPrune)
    RL = RLGrow (PosGrow, NegGrow)
    RL = RLPrune (RL, PosPrune, NegPrune)
    add RL to RS
    IF Desc_length (RS, Pos, Neg) > DL_LENGTH+64 THEN
        // For pruning the entire rule set. Exit when done
        FOREACH RL R in RS
            IF Desc_length (RS -> R, Pos, Neg) < DL_LENGTH THEN
                remove R from RS
                DL_LENGTH = Desc_length (RS, Pos, Neg)
            ENDIF
        ENDFOR
       return (RS)
    ENDIF
    DL_LENGTH = Desc_length (RS, Pos, Neg)
    remove from Pos and Neg all instances covered by RL
    ENDWHILE
End BUILDRS

Module OPTIMIZERS (RS, Pos, Neg)
    FOREACH RL R in RS
        remove R from RS
        U Posval = instances in Pos uncovered by RS
        U Negval = instances in Neg uncovered by RS
        spilt (U Posval, U Negval) into (PosGrow, NegGrow) and (PosPrune, NegPrune)
        RepRL = RLGrow (PosGrow, NegGrow)
        RepRL = RLPrune (RepRL, PosPrune, NegPrune)
        RevRL = RLGrow (PosGrow, NegGrow, R)
        RevRL = RLPrune (RevRL, PosPrune, NegPrune)
        choose better of RepRL and RevRL and add to RS
    ENDFOR
End OPTIMIZERS

Module RIPPER (Pos,Neg, n)
    RS = BUILDRS (Pos,Neg)
    repeat n times RS = OPTIMIZERS (RS, Pos, Neg)
    return (RS)
End RIPPER
```

PART

Projective Adaptive Resonance Theory (PART) employed partial decision tree approach to infer rules. The specific characteristic of PART is that the algorithm does

not need to carry out global optimization strategy as in the case of RIPPER and C4.5 in order to produce the appropriate rules [22, 23]. The algorithm description is as follows:

Algorithm 2. PART rule induction algorithm

Inputs: Dataset S,
F1 -> dimensions of input vectors,
F2 -> expected maximum clusters allowable at each clustering level
Initial parameters $\rho_0, \rho_h, \sigma, \alpha, \theta,$ *and* e.
Output: RuleSet -> set of rules

Let $\rho = \rho_0$.
L1: WHILE (not stopping condition i.e stable clusters not yet formed)
FOREACH input vector in S do
Calculate hij for all F1 nodes Vi and committed F2 nodes Vj. If all F2 nodes are non-committed, goto L2
Calculate Tj for all committed F2 nodes Vj.
L2: Select the best F2 node Vj. If no F2 node can be picked, add the input data into outlier O and then proceed with L1
If the best is a committed node, calculate rj, else goto L3
If rj >= ρ, goto L3, else reset the best Vj and goto L2
L3: Set the winner Vj as the committed and update the bottom-up and top-down weights for winner node Vj.
ENDFOR

FOREACH cluster Cj in F2, calculate the associated dimension set Dj. Then, let S = Cj
$\rho = \rho + \rho_h$, then, goto L1.
For the outlier O, let S = 0, goto L1
ENDWHILE

Hybrid Rule Induction Algorithm

The proposed hybrid rule based algorithm leverages the capabilities of JRip and PART algorithms to generate decision rules for detecting phishing URL. The hybrid algorithm is described as follow:

Algorithm 3. Proposed hybrid rule based algorithm

Inputs: Dataset *S*, with Pos and Neg instances
F1, *F2*, ParameterList
Output: RuleSet -> set of rules
JRipRuleSet = JRip(Pos,Neg)
PARTRuleSet = PART(S,F_1,F_2,ParameterList)
HybridRuleSet = JRipRuleSet U PARTRuleSet
HybridRuleSet = RemoveDuplicateRules(HybridRuleSet)
return (HybridRuleSet)

3.4 Evaluation Metrics

The study employs standard evaluation metrics to ascertain the performance of the proposed approach. These metrics include the total number of rules generated by each

rule induction algorithm, accuracy, Kappa statistics, Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). Accuracy, Kappa, MAE, and RMSE are calculated using the Eqs. 1, 2, 3 and 4 respectively. The number of correctly classified phishing URLs denotes True Positive (TP) while the number of correctly classified legitimate URLs represents True Negative (TN). False Positive (FP) denotes the number of legitimate URLs that were identified as phishing and False Negative (FN) denotes the number of phishing URLs identified as legitimate links. In Kappa statistic calculation, Po and Pe are the probability of observed and expected agreement respectively. MAE is calculated by dividing the sum of absolute errors by the number of samples used during the training stage and similarly, RMSE is computed as shown in Eq. 4.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Kappa = \left(\frac{P_o - P_e}{1 - P_e}\right) \tag{2}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |e_i| \tag{3}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2} \tag{4}$$

4 Results and Discussion

To evaluate the performance of the hybrid model for phishing URL detection, different experiments were conducted using PhishingDataset1 and PhishingDataset2 respectively. All experiments were conducted using R statistical package and RWeka library. R is an open source high-level programming language and software development environment that is widely used for algorithm implementation, data analysis, model development and numerical computation. The implementation of the rule induction algorithms was carried out on Windows 8 operating system. The system has a random access memory (RAM) of 4 GB and 2.40 GHz Intel Core i3 CPU with 1 TB Hard Disk. Cross-validation based on 10-fold was utilized to check the behaviors of the selected rule induction algorithms across the different phishing datasets.

4.1 Classification Performance Based on PhishingDataset1

This section discusses the results of the rule induction algorithms based on PhishingDataset1. As shown in Fig. 2, the number of rules generated by PART algorithm is more than the JRip. PART rule induction algorithm produced 41 rules based on PhishingDataset1 while JRip produced 15 rules. The proposed hybrid rule induction algorithm produced 55 rules. The top 10 rules generated by JRip and PART algorithms

are shown in Figs. 3 and 4 respectively. From these tables, rule 7 of JRip and rule 3 of PART are the same. The proposed hybrid rule induction algorithm removed duplicate rules from the two algorithms to obtain unique rule set.

Table 3 shows that PART algorithm outperformed JRip according the results obtained during the experiment on PhishingDataset1. Based on the standard evaluation metrics employed in this study, PART produced accuracy, Kappa, MAE, and RMSE of 0.9364, 0.8874, 0.0689, and 0.1855 respectively as compared to JRip rule induction algorithm with 0.9239, 0.8656, 0.0882, and 0.21 respectively. These results demonstrate the superiority of PART algorithm over JRip for phishing URL detection. However, as shown in Fig. 5 the proposed hybrid rule induction algorithm outperformed PART algorithm based on accuracy considered for performance comparison. The hybrid rule induction algorithm achieved accuracy of 0.9453.

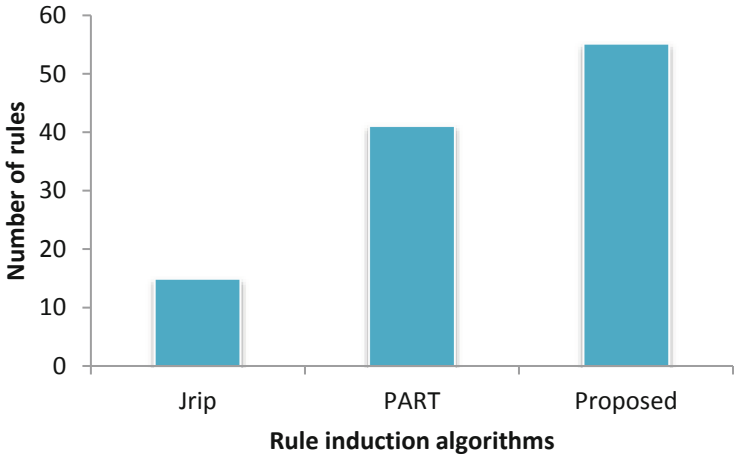

Fig. 2. Number of rules generated by the rule induction algorithms based on PhishingDataset1

1. IF (Request_URL = -1) AND (URL_Length = 1) AND (URL_of_Anchor = -1) AND (SSLfinal_State = 1) => Result=0
2. IF (SFH = -1) AND (URL_of_Anchor = 1) AND (SSLfinal_State = 1) AND (Request_URL = -1) => Result=0
3. IF (SSLfinal_State = 0) AND (URL_of_Anchor = 0) AND (SFH = 0) => Result=0 (14.0/0.0)
4. IF (Request_URL = -1) AND (URL_Length = 1) AND (SSLfinal_State = -1) AND (URL_of_Anchor = 1) => Result=0
5. IF (SSLfinal_State = 0) AND (web_traffic = -1) AND (SFH = -1) AND (URL_of_Anchor = -1) => Result=0
6. IF (Request_URL = -1) AND (URL_of_Anchor = 0) AND (SFH = 1) AND (SSLfinal_State = 1) => Result=0
7. IF (SFH = -1) AND (URL_of_Anchor = -1) AND (SSLfinal_State = -1) => Result=1
8. IF (popUpWidnow = -1) AND (SFH = -1) => Result=1
9. IF (SFH = 0) => Result=1
10. IF (SFH = -1) AND (Request_URL = -1) => Result=1

Fig. 3. Top 10 rules generated by JRip based on PhishingDataset1

1. IF SFH = 0 AND Request_URL = 0 AND URL_of_Anchor = 1 => Result=1
2. IF SFH = 1 AND SSLfinal_State = 1 AND URL_of_Anchor = 1 => Result= -1
3. IF SFH = -1 AND URL_of_Anchor = -1 AND SSLfinal_State = -1 => Result=1
4. IF SFH = 0 AND SSLfinal_State = -1 => Result= 1
5. IF SFH = 1 AND popUpWidnow = 1 AND URL_Length = 0 => Result= -1
6. IF SFH = 1 AND popUpWidnow = 0 AND Request_URL = 0 => Result= -1
7. IF SFH = -1 AND popUpWidnow = 1 AND Request_URL = 0 => Result= -1
8. IF SFH = -1 AND Request_URL = 1 AND popUpWidnow = -1 => Result=1
9. IF SFH = -1 AND Request_URL = 1 AND age_of_domain = 1 => Result= -1
10. IF SFH = -1 AND URL_of_Anchor = 0 AND Request_URL = -1 => Result= 1

Fig. 4. Top 10 rules generated by PART based on PhishingDataset1

Table 3. Performance evaluation of JRip and PART on PhishingDataset1

	Algorithm		
	JRip	PART	Proposed
Accuracy	0.9239	**0.9364**	**0.9453**
Kappa	0.8656	**0.8874**	
MAE	0.0882	**0.0689**	
RMSE	0.21	**0.1855**	

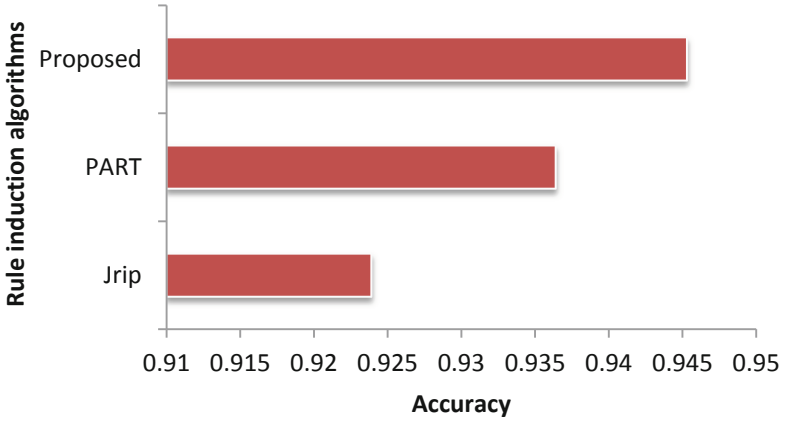

Fig. 5. Accuracy of the rule induction algorithms on PhishingDataset1

4.2 Classification Performance Based on PhishingDataset2

This section presents the results of the rule induction algorithms based on PhishingDataset2. As discussed in Sect. 3.1, PhishingDataset2 is a dataset containing 11,055 samples, which is larger than the instances in PhishingDataset1. Similarly, according to

the results in Fig. 6, PART algorithm produced more rules than the JRip algorithm. 163 rules were generated from PART algorithm while JRip produces 30 rules. The proposed hybrid rule-based algorithm generated 191 rules. The top 10 rules produced by JRip and PART based on PhishingDataset2 are shown in Figs. 7 and 8 respectively.

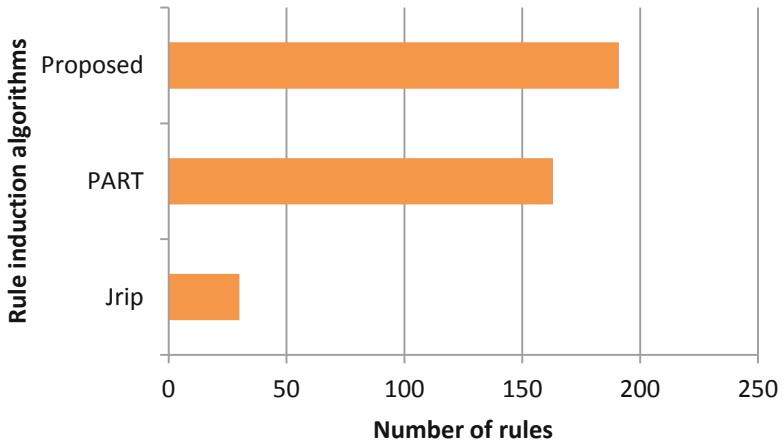

Fig. 6. Number of rules generated by the rule induction algorithms based on PhishingDataset2

1. IF (URL_of_Anchor = -1) AND (SSLfinal_State = -1) => Result=-1
2. IF (URL_of_Anchor = -1) AND (SSLfinal_State = 0) => Result=-1
3. IF (SSLfinal_State = -1) AND (Links_in_tags = -1) => Result=-1
4. IF (web_traffic = 0) AND (having_Sub_Domain = -1) AND (age_of_domain = -1) AND (DNSRecord = -1) => Result=-1
5. IF (web_traffic = 0) AND (URL_of_Anchor = -1) => Result=-1
6. IF (SSLfinal_State = -1) AND (Domain_registeration_length = 1) AND (SFH = -1) AND (Links_in_tags = 1) AND (Links_pointing_to_page = 1) => Result=-1
7. IF (web_traffic = 0) AND (URL_of_Anchor = 0) AND (having_Sub_Domain = 0) AND (having_IP_Address = -1) AND (Links_pointing_to_page = 0) => Result=-1
8. IF (SSLfinal_State = 0) => Result=-1
9. IF (SSLfinal_State = -1) AND (Domain_registeration_length = 1) AND (Links_in_tags = 0) AND (Links_pointing_to_page = 0) => Result=-1
10. IF (web_traffic = 0) AND (URL_of_Anchor = 0) AND (having_Sub_Domain = 0) AND (Links_in_tags = 1) => Result=-1

Fig. 7. Top 10 rules generated by JRip based on PhishingDataset2

1. IF (SSLfinal_State = 0) AND (URL_of_Anchor = -1) => Result= -1
2. IF (SSLfinal_State = 0) AND (Links_pointing_to_page = 1) => Result= -1
3. IF (SSLfinal_State = 1) AND (URL_of_Anchor = 1) AND (Google_Index = 1) AND (having_IP_Address = 1) => Result= 1
4. IF (SSLfinal_State = 1) AND (URL_of_Anchor = 1) AND (Request_URL = 1) => Result= 1
5. IF (SSLfinal_State = -1) AND (Prefix_Suffix = -1) AND (URL_of_Anchor = -1) => Result= -1
6. IF (SSLfinal_State = 1) AND (URL_of_Anchor = 1) AND (Links_pointing_to_page = 1) AND (having_IP_Address = -1) => Result= 1
7. IF (SSLfinal_State = 1) AND (URL_of_Anchor = 0) AND (web_traffic = -1) AND (Google_Index = 1) => Result= 1
8. IF (SSLfinal_State = 0) AND (Links_pointing_to_page = 0) AND (having_Sub_Domain = 0) => Result= -1
9. IF (SSLfinal_State = 1) AND (URL_of_Anchor = 0) AND (web_traffic = 1) AND (SFH = 1) => Result= 1
10. IF (Prefix_Suffix = 1) => Result= 1

Fig. 8. Top 10 rules generated by PART based on PhishingDataset2

According to the results in Table 4, PART rule induction algorithm outperformed JRip with accuracy, Kappa, MAE, and RMSE of 0.9823, 0.964, 0.0281, 0.1185 respectively while JRip algorithm produces accuracy, Kappa, MAE, and RMSE of 0.9547, 0.908, 0.0825, 0.2031 respectively. These results further guaranteed the suitability of the proposed hybrid rule induction algorithm for detecting phishing URL which achieved accuracy of 0.9908 on PhishingDataset2. Figure 9 shows the performance accuracy of the three rule induction algorithms investigated in this research.

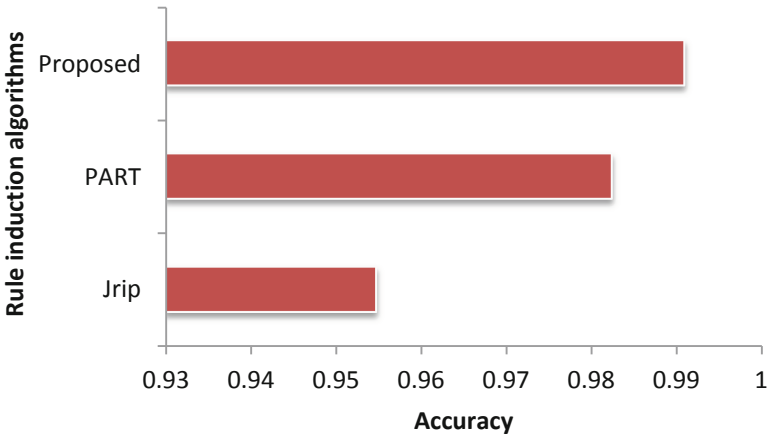

Fig. 9. Accuracy of the rule induction algorithms on PhishingDataset2

Table 4. Performance evaluation of JRip and PART on PhishingDataset2

	Algorithm		
	JRip	PART	Proposed
Accuracy	0.9547	**0.9823**	**0.9908**
Kappa	0.908	**0.964**	
MAE	0.0825	**0.0281**	
RMSE	0.2031	**0.1185**	

5 Conclusion

Phishing detection has been a major challenge to Internet users and the entire World Wide Web (WWW) community at large. A number of strategies based on social engineering have been deployed by attackers to successfully launch phishing attack. This paper explored the possibility of detecting phishing attack at early stage using a combination of rules generated from two widely used rule induction algorithms: JRip and PART. The results of the various experiments conducted indicated that PART algorithm is superior to JRip when it comes to phishing detection problem. Based on two publicly available datasets for phishing detection, PART algorithm produces promising results in terms of the standard performance metrics employed in this study based on accuracy, Kappa, MAE, and RMSE. Therefore, by extension, these results impacted positively on the proposed hybrid rule induction algorithm which fused the rules from the two selected rule induction algorithms. Thus, the hybrid algorithm proposed in this study outperformed both JRip and PART in terms of accuracy. In future, the authors intend to explore phishing detection using adaptive machine learning methods to address zero-day phishing attack.

References

1. Abdelhamid, N., Ayesh, A., Thabtah, F.: Phishing detection based associative classification data mining. Expert Syst. Appl. **41**(13), 5948–5959 (2014)
2. Dhamija, R., Tygar, J.D., Hearst, M.: Why phishing works. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM (2006)
3. He, Q., Ma, X.: A large-scale URL filtering algorithm in high-speed flow. In: Proceedings of 2016 2nd IEEE International Conference on Computer and Communications, ICCC 2016 (2017)
4. Manan, W.N.W., Ahmed, A.G.A., Kahar, M.N.M.: Characterizing current features of malicious threats on websites. In: Vasant, P., Zelinka, I., Weber, G.W. (eds.) ICO 2018. AISC, vol. 866, pp. 210–218. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-00979-3_21
5. Jayakanthan, N., Ramani, A.V., Ravichandran, M.: Two phase classification model to detect malicious URLs. Int. J. Appl. Eng. Res. **12**(9), 1893–1898 (2017)
6. Vanhoenshoven, F., et al.: Detecting malicious URLs using machine learning techniques. In: 2016 IEEE Symposium Series on Computational Intelligence, SSCI 2016 (2017)

7. Gupta, S.: Efficient malicious domain detection using word segmentation and BM pattern matching. In: 2016 International Conference on Recent Advances and Innovations in Engineering, ICRAIE 2016 (2017)
8. Thakur, S., Meenakshi, E., Priya, A.: Detection of malicious URLs in big data using RIPPER algorithm. In: Proceedings of RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology (2018)
9. Liu, J., et al.: A Markov detection tree-based centralized scheme to automatically identify malicious webpages on cloud platforms. IEEE Access **6**, 74025–74038 (2018)
10. Vijayarani, S., Divya, M.: An efficient algorithm for generating classification rules. Int. J. Comput. Sci. Technol. **2**(4), 512–515 (2011)
11. Grzymala-Busse, J.W.: Rule induction. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 249–265. Springer, Boston (2010). https://doi.org/10.1007/978-0-387-09823-4_13
12. Lee, S., Kim, J.: Warning bird: a near real-time detection system for suspicious URLs in Twitter stream. IEEE Trans. Dependable Secur. Comput. **10**(3), 183–195 (2013)
13. Adewole, K.S., et al.: SMSAD: a framework for spam message and spam account detection. Multimedia Tools Appl. **78**, 3925–3960 (2017)
14. Adewole, K.S., et al.: Twitter spam account detection based on clustering and classification methods. J. Supercomput. 1–36 (2018)
15. Bhardwaj, T., Sharma, T.K., Pandit, M.R.: Social engineering prevention by detecting malicious URLs using artificial bee colony algorithm. In: Pant, M., Deep, K., Nagar, A., Bansal, J.C. (eds.) Proceedings of the Third International Conference on Soft Computing for Problem Solving. AISC, vol. 258, pp. 355–363. Springer, New Delhi (2014). https://doi.org/10.1007/978-81-322-1771-8_31
16. Darling, M., et al.: A lexical approach for classifying malicious URLs. In: Proceedings of the 2015 International Conference on High Performance Computing and Simulation, HPCS 2015 (2015)
17. Xuan, J., Yongzhen, L.: The Detection method for two-dimensional barcode malicious urls based on the hash function. In: Proceedings of 2016 3rd International Conference on Information Science and Control Engineering, ICISCE 2016 (2016)
18. Dewan, P., Kumaraguru, P.: Facebook Inspector (FbI): Towards automatic real-time detection of malicious content on Facebook. Soc. Netw. Anal. Min. **7**(1), 15 (2017)
19. Mohammad, R.M., Thabtah, F., McCluskey, L.: An assessment of features related to phishing websites using an automated technique. In: 2012 International Conference for Internet Technology and Secured Transactions. IEEE (2012)
20. Veeralakshmi, V., Ramyachitra, D.: Ripple Down Rule learner (RIDOR) classifier for IRIS dataset. IJCSE **1**(1), 79–85 (2015)
21. Cohen, W.W.: Fast effective rule induction. In: Proceedings of the Twelfth International Conference on Machine Learning (1995)
22. Ali, S., Smith, K.A.: On learning algorithm selection for classification. Appl. Soft Comput. **6**(2), 119–138 (2006)
23. Frank, E., Witten, I.H.: Generating accurate rule sets without global optimization (1998)