Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection

Mahdieh Zabihimayvan and Derek Doran
Department of Computer Science and Engineering
Wright State University, Dayton, OH, USA
{zabihimayvan.2, derek.doran}@wright.edu

Abstract—**Phishing as one of the most well-known cybercrime activities is a deception of online users to steal their personal or confidential information by impersonating a legitimate website. Several machine learning-based strategies have been proposed to detect phishing websites. These techniques are dependent on the features extracted from the website samples. However, few studies have actually considered efficient feature selection for detecting phishing attacks. In this work, we investigate an agreement on the definitive features which should be used in phishing detection. We apply Fuzzy Rough Set (FRS) theory as a tool to select most effective features from three benchmarked data sets. The selected features are fed into three often used classifiers for phishing detection. To evaluate the FRS feature selection in developing a generalizable phishing detection, the classifiers are trained by a separate out-of-sample data set of 14,000 website samples. The maximum F-measure gained by FRS feature selection is 95% using Random Forest classification. Also, there are 9 universal features selected by FRS over all the three data sets. The F-measure value using this universal feature set is approximately 93% which is a comparable result in contrast to the FRS performance. Since the universal feature set contains no features from third-part services, this finding implies that with no inquiry from external sources, we can gain a faster phishing detection which is also robust toward zero-day attacks.**

Index Terms—**Phishing Detection, Fuzzy Rough Set, Feature Selection**

I. Introduction

Phishing is categorized as a social engineering attack to coheres users to perform adverse actions on behalf of an attacker [1]. A usual purpose of such attacks is to deceive users to steal their personal or confidential information. To this end, phishers try to mimic the emails or web pages that indicate a high visual similarity to the legitimate emails or web pages. The pages are further accompanied with at least one login form to gather personal or confidential user information [2]. The content of phishing emails and web pages are developed from social engineering practices so that victims are persuaded to follow the instructions with the hope of, for example, updating or validating their information, finding a job, winning a big prize, or receiving a significant discount to a service. According to the most up-to-date trend reports by the Anti-Phishing Working Group [3], over 151,014 unique phishing websites and 270,557 unique emails have been documented. These sites and emails mimicked at least 777 brands targeted by phishing campaigns during the third quarter of 2018.

A range of technical strategies have been proposed to detect phishing emails and websites. The strategies can broadly be categorized into three classes: (i) list-based; (ii) heuristic; and (iii) and machine learning (ML)-based approaches [4]. The first class simply references different lists of legitimate and phishing URLs. While these lists are frequently updated, there is an inevitable time gap between the emergence of a phishing website, its observation, reporting, and finally, addition to a suspicious URL list. This makes list-based strategies vulnerable towards emerging and zero-day phishing attacks [5]. Heuristic approaches can automatically recognize a phishing web pages based a suite of features extracted from them [6]. One method, for example, focuses on the similarity of the suspicious web page with the legitimate one based on visual features such as image, logo, and textual content of the page [7]. However, there is no guarantee that a phishing site and a legitimate web page or email has features in common, which can cause poor detection accuracy in practice. Furthermore, heuristic techniques can be bypassed by a knowledgeable attacker that ensures their phishing website does not contain the detected features. ML methods [8] are also dependent on the set of features extracted for each web page and further require ground truth phishing and legitimate websites for training. The quality and variety of websites in this training set has a strong effect on the final detection accuracy [9], and it can be expensive to obtain a training set that is suitably sized and diverse.

Despite these challenges, ML approaches for phishing detection have been an active area of research. Several studies have been conducted on different data sets using various classification algorithms [10]–[12]. There is an undeniable effect of the features used for classification on the accuracy of an algorithm [13], but few studies have actually considered intelligent methods feature selection. The problem of feature selection is crucial to build phishing detection systems that are generalizable in practice. For example, it is feasible that the set of features best distinguishing phishing websites that imitate a financial institution may be different from sites imitating an e-commerce platform, or of websites that are encoded in different languages. The present art in feature selection are simply based on heuristic thresholds [13], [14].

To address this problem, we apply Fuzzy Rough Set (FRS) theory [15] as a tool to select the most effective features for detecting phishing websites in this paper. Our feature selection scheme preprocesses features that are fed into three often used classification methods for phishing detection: a multilayer

perceptron, a random forest, and SMO. We train each classifier on a separate out-of-sample dataset to examine the ability of FRS feature selection in developing a generalizable phishing detection. This training set which is composed by random extraction of websites from an online repository [16] contains 7,000 legitimate and 7,000 phishing website samples. The benefit of using FRS for feature selection is realized by comparing our results against three recently proposed algorithms [13], [14].

The rest of this paper is organized as follows: Section II discusses the related work on phishing detection. Section III presents the procedure used for feature selection. Section IV evaluates the proposed method and compares its performance with other related algorithms. Finally, Section V summarizes the main conclusions.

II. RELATED WORK

ML-based approaches to detect phishing websites is an active research area that employs a wide range of supervised classification techniques to segregate phishing class. Feng *et al.* propose a novel neural network for phishing detection [10]. They improve the generalization ability of the network by designing risk minimization principle. Performance of the proposed network is evaluated over a UCI repository [1] containing 11,055 samples labeled as phishing/legitimate. The dataset also specifies 30 features for each website categorized as Address bar-based, Abnormal-based, HTML/Javascript-based, and Domain-based features. Rao and Pais propose a novel algorithm to detect phishing websites using both machine learning techniques and image checking. They also extract features from URL, website content, and third-party services [9]. It is worth mentioning that although using the features from third-party service can increase the detection time, it increases the detection accuracy in practice [12]. They evaluate the performance of the proposed algorithm over 1407 legitimate and 2119 phishing websites collected from PhishTank [2] and Alexa database [3], respectively.

Mohammad *et al.* propose a novel self-structuring neural network for detecting phishing websites [4]. They specify 17 features, some extracted from third-party service, for 600 legitimate and 800 phishing websites collected from PhishTank and Millersmiles [4] archives. Their experiments indicate the high generalizability and ability of the neural network in phishing detection. In another work, they propose a feed forward neural network trained by back propagation to classify websites [11]. 18 features are specified for 859 legitimate and 969 phishing websites, respectively. Jain and Gupta propose a machine learning-based technique using only client-side features to detect phishing websites [17]. They extract 19 features from the URL and source code of the web pages and evaluate their method over 2,141 phishing web pages from

PhishTank and Openfish [5] and 1,918 legitimate pages from Alexa database, some online payment, and banking websites.

Although all the above-mentioned studies have suggested different features to detect phishing websites, some features may not be sufficiently discerning to highlight phishing instances [13]. Only a limited number of work has focused on selecting the most effective features to detect phishing websites. Rajab proposes using Correlation Feature Set (CFS) and Information Gain (IG) to select the most influential features to detect phishing activities [13]. The results over the UCI repository containing 30 features specified for 11,055 samples indicate 11 and 9 features selected by IG and CFS, respectively. The classification performance using selected features is also evaluated by a data mining method called RIPPER. Similarly, Babagoli *et al.* use a similar data set and propose feature selection using decision trees and the wrapper method [18] which results in selecting 20 features [14]. They evaluate the phishing detection performance using a novel meta-heuristic-based nonlinear regression algorithm. Still, the feature selection methods proposed by these studies are dependent on the data and requires user-specified threshold values which should be set heuristically. These thresholds can affect the ultimate performance of the classification algorithm, especially when features are selected based on out-of-sample training data in practice.

In this work, we propose using FRS theory as the feature selection algorithm. In contrast to the related work focusing on only one data set, our experiments evaluate the generalizability of our approach by finding a universal set of discriminative features from three benchmark data sets. The datasets include a new source of 7,000 legitimate and 7,000 phishing website samples randomly extracted from the a phishing website repository [16]. We compare the detection performance using the selected features with the results of features selections proposed in [13] [14].

III. FUZZY ROUGH SET FEATURE SELECTION

Rough Set (RS) theory, developed by Pawlak [15], is a method to decide how to separate a set of data, with a decision boundary and an indiscernibility relation R. Supposing each set member s_i described by D features with discrete values, R defines a binary similarity between two members s_m and s_n as follows:

$$R(s_m, s_n) = \begin{cases} 1 & \text{if } \forall\, i \in \{1, ..., D\} \;\; s_m(i) = s_n(i) \\ 0 & \text{if } \exists\, i \in \{1, ..., D\} \text{ where } s_m(i) \neq s_n(i) \end{cases}$$

Similar samples join individually together in one exclusive class, $[s_i]_R$, which is defined as follows:

$$[s_i]_R = \{s_j \mid R(s_i, s_j) = 1\}$$

To better understand the definition of $[s_i]_R$, figure 1 illustrates a universe of discourse as a grid of equivalence class (squares) representing set members that have identical feature values ($R = 1$). The closed curved line further defines a hypothetical
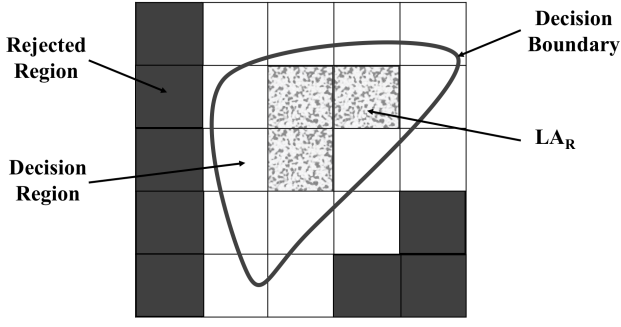
Fig. 1: The universe of discourse regions segregated by RS

decision boundary. The dotted squares show regions that are fully located within the decision boundary. Such regions are equivalence classes are fully describable by the boundary and are called the boundarys lower approximation set LA_R. The boundary region also contains the sets of equivalence classes (white squares) that are both inside and outside of the decision boundary. This boundary region unioned with LA_R forms the boundarys upper approximation set UA_R. Finally, the dark squares lying outside of the decision boundary specify the samples that are rejected by the decision boundary. Supposing f is the function defining the decision boundary and U indicating the universe of discourse, UA_R and LA_R are declared as:

$$LA_R = \{[s_i] \in U \mid \forall s_j \in [s_i]_R : f(s_j) \leq 0\}$$

$$UA_R = \{[s_i] \in U \mid \exists s_j \in [s_i]_R : f(s_j) \leq 0\}$$

For features with continuous values, RS is extended by Fuzzy Rough Set (FRS) [19] theory. FRS asserts that there is a subset of features representing the *labels* of a set member while the remaining features are non-label. Assuming the feature values are normalized between 0 and 1, the similarity relation between two set members R is given by:

$$\forall i \in F, L : R(s_m, s_n) = \tau(\{max(0, 1 - \Delta_2(s_m(i), s_n(i)))\})$$

where F and L are the sets of non-label and label features, $s_m(i)$ shows the value of ith feature of sample s_m, and $\Delta_2(x, y)$ defines the squared difference between x and y. τ is the Lukasiewicz t-norm [20] that aggregates the similarity of all the features of the two samples and returns a values in $[0, 1]$. The equation below indicates how to calculate the τ where x and y are both in $[0, 1]$:

$$\tau(x, y) = max(0, x + y - 1)$$

To determine if a set member s_m is in the upper or lower approximation set of a decision boundary, FRS defines Upper and Lower Approximation Memberships μ_U and μ_L as:

$$\mu_U(s_m) = \sup_{s_n \in U, s_n \neq s_m} \tau(R_F(s_m, s_n), R_L(s_m, s_n))$$

$$\mu_L(s_m) = \inf_{s_n \in U, s_n \neq s_m} I(R_F(s_m, s_n), R_L(s_m, s_n))$$

TABLE I: Data set description

Dataset	Year	Size	Num. Features	Num. Phishing Sites
UCI1	2014	11,055	30	3,793
Mendeley	2018	10,000	48	5,000
UCI2	2014	1,353	9	702

TABLE II: Feature classes across the three data sets

Feature Class	Description
Address bar-based	Features of the web page URL such as its length in character, port, and the protocol used to transfer information.
Abnormal-based	Features of abnormal activities on a page, such as loading embedded objects from external domains or submissions to an email address.
HTML/Javascript-based	Features of HTML and Javascript functions embedded in the source code of the page.
Domain-based	Features related to information from third-party services.

where $R_F(s_m, s_n)$ shows the similarity between s_m and s_n using only non-label features while $R_L(s_m, s_n)$ gives the similarity using only label features set. I is the fuzzy Implicator operator that is declared as follows where again $q, s \in [0, 1]$:

$$I(q, s) = min(1, 1 - q + s)$$

A higher value of $\mu_L(s_m)$ gives us more confidence that s_m is fully describable by the decision boundary. The values greater than zero for this membership function are considered for feature selection used in this work.

IV. EXPERIMENTS

We apply FRS to select features from three benchmark data sets for phishing attacks. Table I indicates the description of data used in this work. The first data set is from the UCI repository, UCI1[6], that contains 30 features over 11,055 samples labeled as phishing/legitimate websites. Based on [4], the proposed features of this data set are categorized into 4 groups: Address bar-based, Abnormal-based, HTML/Javascript-based, and Domain-based features. Table II presents a description of these classes in more detail. The second data set, Mendeley [21], specifies 48 features for 5,000 legitimate and 5,000 phishing websites. According to the feature class descriptions in Table II, these 48 features can be also categorized into three groups of Address bar-based, Abnormal-based, and HTML/Javascript-based features. Also, Table III represents 23 features of this data set which are shared with the features in UCI1. The UCI2 dataset[7] specifies 9 features from across the four feature classes over 702 phishing, 548 legitimate, and 103 suspicious websites. Suspicious website has features of both legitimate and phishing samples and thus, can be considered as either phishy or legitimate [22].

The differences in sample size and number of features among the three data sets indicate some differences in the characteristics of the data studied in this work. In UCI1,

[6]https://archive.ics.uci.edu/ml/datasets/phishing+websites
[7]https://archive.ics.uci.edu/ml/datasets/Website+Phishing

the number of legitimate websites is more than twice of the number of phishing samples while in Mendeley, these classes have equal population size. In UCI2, although the sample sizes of both classes are approximately similar and have the dominant population, there is a minority class for suspicious samples. Furthermore, in contrast to the UCI repositories published in 2014, the Mendeley data has been published recently and thus, can be a representative of more updated samples for phishing websites. Finally, in contrast to some related work focusing on small number of data [4], [9], [11], [23], the sample size of data sets we consider is notably high.

To evaluate the detection performance using features selected by FRS, we employ three well-known classification algorithms: the multilayer-perceptron, random forest, and sequential minimal optimization (SMO) that have been extensively employed for phishing detection by other related studies [9], [12], [13], [17]. All hyper-parameters are also set based on the values used in these previous works. To evaluate the ability for FRS feature selection to develop a generalizable phishing detector, we train each classifier on a separate out-of-sample dataset composed by random extraction of phishing and non-phishing websites from an online repository [16]. This training set has 7,000 legitimate and 7,000 phishing website samples.

We compare the results using FRS against three recently proposed feature selection algorithms: information gain (IG), correlated feature set [13] (CFS), and using a combination of decision tree and the Wrapper method [14] (DW). IG chooses features from a ranking based on the level of information each provides with respect to the target class (features which specifies the class labels) [13]. CFS sorts features based on their correlation with the target class. The first-ranked feature has highest correlation with the target class and lowest correlation with the others [13]. DW utilizes a combination of decision tree and Wrapper to select an optimal feature set [14]. Since IG and CFS just provide a ranking of features, a threshold is needed to specify which features should be selected as efficient. Furthermore, DW is dependent on the expert opinion to decide what decreasing in the accuracy should be considered as significant to stop the method. In this work, the thresholds and other settings are set based on the values suggested in [13] and [14].

Figure 2 presents the F-measure values for classifiers using four feature selection algorithms over each data set. Assuming phishing and legitimate samples belong to positive and negative classes respectively, F-measure calculates the quality of classification for both classes [24]. This metric is defined as $2 \times \frac{Precision \times Recall}{Precision + Recall}$ where $Precision = \frac{TP}{TP+FP}$ indicates number of truly detected phishing samples over number of all samples identified as phishing. Also, $Recall = \frac{TP}{TP+FN}$ is the number of truly identified phishing samples over the number of all existed phishing websites.

Across all datasets, the random forest model yields the best performance no matter the feature selection method considered. Over UCI1 and Mendeley, the FRS performance is higher than the other feature selection algorithms. The

TABLE III: Features shared between UCI1 and Mendeley

No.	Feature Name	Type	Description
1	UrlLen	D	length of URLs.
2	ExtFavicon	B	Favicon is from a different domain.
3	NumDash	D	No. dash in webpage URL.
4	NumDashInHostname	D	No. dash in URL host name.
5	AtSymbol	B	@ exists in the URL.
6	NoHttps	B	HTTP exists in the URL.
7	IpAddress	B	URL host name contains IP.
8	DomainInPaths	B	URL path contains TLD/ccTLD.
9	UrlLenRT	C	No. all characters in the webpage URL.
10	PctExtResUrls	C	% external resource URLs in the HTML source code.
11	PctExtHlinks	C	% external links in the HTML source code.
12	ExtFormAct	B	An external domain URL in the form action attribute.
13	AbnormFormAct	C	Occurring abnormal strings (#, about:blank, empty string, javascript:true) in form action.
14	PctNullSelfRedirHlinks	C	% hyperlink fields of current URL, empty, abnormal, or self-redirect values.
15	Submit2Email	B	HTML source code contains mailto.
16	PctExtResUrlsRT	C	% external resource URLs in webpage HTML source code.
17	AbnormExtFormActR	C	Occurring empty string, foreign domain, about:blank in form action.
18	ExtMetaScriptLinkRT	C	% meta, script and link tags containing external URLs.
19	PctExtNullSelfRedirHlinksRT	C	% hyperlinks in HTML source code with different domain names.
20	RightClickDisabled	B	HTML source code contains command to disable right click.
21	PopUpWin	B	HTML source code contains pop-ups.
22	FakeLinkInStatusBar	B	HTML source code contains onMouseOver.
23	IframeOrFrame	B	HTML source code contains frame/iframe.

The column Type indicates the value types of features. D, B, and C show discrete, binary, and categorical values, respectively.

identical performance of FRS and IG over UCI2 is also the result of both methods selecting all features, which could be a byproduct of the small size of this dataset. The small number of features on UCI2 also contribute to the comparatively low performance of all classifiers on this dataset. In contrast to IG, CFS, and DW which are dependent on some thresholds set heuristically, FRS discovers the data dependencies for feature selection based on the data with no need for additional information. In other words, FRS enables handling incomplete information to select effective features based on fuzzy set
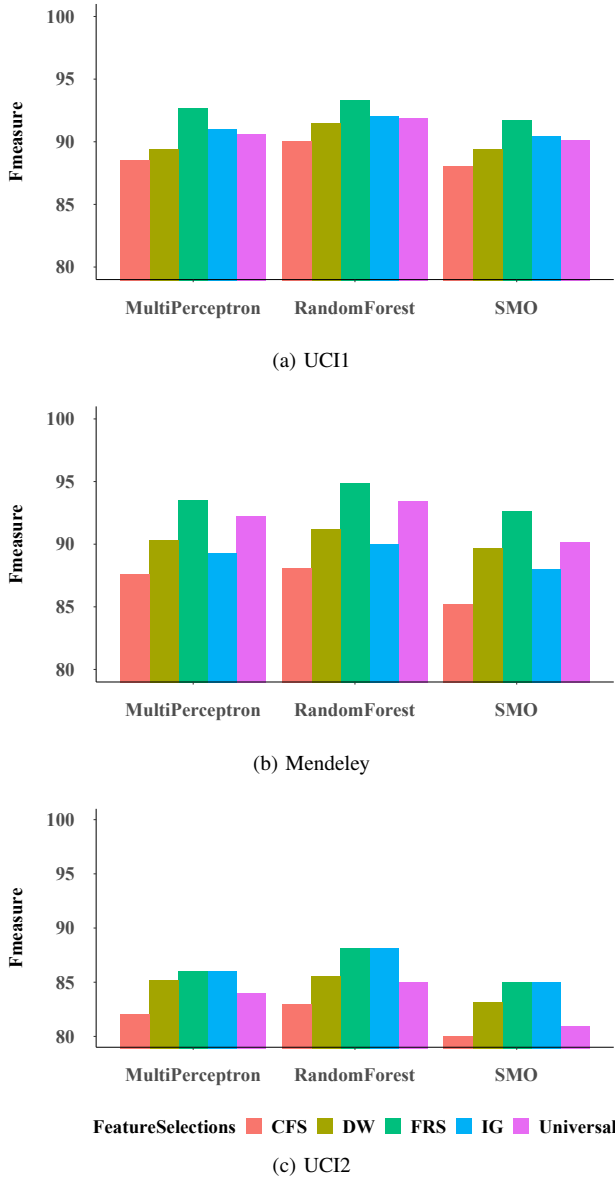
Fig. 2: F-measure for different classifiers and features sets

theory that investigates vagueness in rough set framework [25].

We further investigate the types of features selected by each method. Table IV presents the list of features selected by FRS for each data set. 24, 30, and 9 features have been selected for UCI1, Mendeley, and UCI2, respectively. As indicated, 7 features selected for UCI1 are Domain-based which utilize third-party services such as Alexa or WHOIS databases to get more information about phishing samples. However, Mendeley does not use any Domain-based feature and as Figure 2 indicates, the maximum performance over Mendeley is approximately 95% which outperforms all the FRS results over UCI1. It shows using domain-based features not only increases the detection time due to inquiry from external sources [12], but also has no significant effect to improve the detection performance.

The bold features in Table IV are shared features among all three data sets that have been selected by FRS. FRS identifies the four Address Bar-based features UrlLen, Pref-Suff, HaveSubDomain, and Favicon, accompanied with the five Abnormal-based features of ReqUrl, UrlAnchor, LinksInTags, SFH, and Submit2Email to be "universally applicable" features for phishing detection across all datasets. As phishers often use long URLs, UrlLen is a discrete feature to consider the length of web page URL. Dash in URLs is also another signal for phishing websites which is considered by PrefSuff that checks the existence of prefixes or suffixes separated by dash in the URL. HaveSubDomain examines another way of confusing the website visitors which modifies the number of subdomains in the URL address. Favicon also checks if the favicon indicated in the address bar is loaded from another domain. ReqUrl is an Abnormal-based feature which considers if the external objects within the webpage are loaded from another domain. UrlAnchor checks whether an anchor within the web page has different domain or has no link to other pages. Legitimate websites often use meta, script, and link tags in their HTML document. LinksInTags examines the percentage of such tags in the web page source code. The value of Server Form Handler (SFH) can be another way to distinguish phishing pages. The empty string or "about:blank" in the SFH can indicate legitimate website while different domain name in SFH shows a phishing effort. Finally, using a client-side function called "mailto:" to submit user information to an external server can indicate a phishing attack. This examines by the Boolean feature called Submit2Email.

As mentioned above, the universal features are based on the general signals in the address bar or the HTML document of the web page regardless of its application (e-commerce, online banking, etc), its dominant language, or its statistical reports in external sources like Alexa. The fifth bar for each data set in Figure 2 indicates the performance using only these universal features. Although on UCI2, the low number of features belonging to the universal feature set (only 4) reduces the detection performance, the maximum performance of universal features ($\approx 93\%$) competes with the maximum FRS performance ($\approx 95\%$) over UCI1 and Mendeley. It indicates that although the universal feature set has lower number of features, its F-measure value is comparable with the result of the larger set of features selected by FRS. Since no domain-based feature belongs to the universal feature set, this finding can not only reduce the detection time and make it robust towards zero-day attacks, it can solve the curse of dimensionality for classification techniques [26] [27].

V. CONCLUSION

In this work, we consider finding an agreement about the definitive features which should be used in detecting phishing websites. We evaluate the detection performance using features selected by Fuzzy Rough Set (FRS) theory based on three benchmarked phishing data sets. After selecting the features, we utilize three well-known classification techniques and conduct the experiments by an unseen data set of 14,000 website samples to avoid classification overfitting. Comparisons with

TABLE IV: Features selected by FRS over the data sets

UCI1		Mendeley		UCI2	
Feature	Class	Feature	Class	Feature	Class
IPAddress	1	NumDots	1	IPAddress	1
UrlLen	1	SubDomainLevl	1	**UrlLen**	1
ShortService	1	PathLevl	1	SSLfinalSt	1
AtSymbol	1	**UrlLen**	1	**ReqUrl**	2
PrefSuff	1	**NumDash**	1	**UrlAnchor**	2
HaveSubDomain	1	NumUnderscore	1	**SFH**	2
SSLfinalSt	1	NumQueryComp	1	PopUpWin	3
DomainRegLen	1	NumNumcChars	1	WebTraffic	4
Favicon	1	RandString	1	AgeOfDomain	4
HTTPSToken	1	**DomainInPaths**	1		
ReqUrl	2	HostnameLen	1		
UrlAnchor	2	PathLen	1		
LinksInTags	2	DoubleSlashInPath	1		
SFH	2	**ExtFavicon**	1		
Submit2Email	2	**PctExtResUrls**	2		
Redirect	3	**PctExtHlinks**	2		
PopUpWin	3	InsecureForms	2		
AgeOfDomain	4	RelativeFormAction	2		
DNSRecord	4	**PctNullSelfRedirHlinks**	2		
WebTraffic	4	**Submit2Email**	2		
PageRank	4	FreqDomainNameMismatch	3		
GoogleIndx	4	IframeOrFrame	3		
LinksToPage	4	MissingTitle	3		
StatisticalReport	4	**PctExtResUrlsRT**	3		
		AbnormExtFormActR	3		
		ExtMetaScriptLinkRT	3		
		PctExtNullSelfRedirHlinksRT	3		

Classes 1, 2, 3, and 4 indicate Address bar-based, Abnormal-based, HTML/Javascript-based, and Domain-based feature classes, respectively.

other feature selection methods utilized by the related work indicate the outperformance of FRS in selecting efficient features. The maximum F-measure gained by the features selected by FRS is 95% using Random Forest classification. Also, there are 9 universal features selected by FRS over all the three data sets. The F-measure value using this universal feature set is approximately 93% which is a comparable result in contrast to the FRS performance. Since the universal features contain no domain-based features, this finding implies that with no inquiry from external sources, we can gain a faster phishing detection which is also robust toward zero-day attacks.

VI. ACKNOWLEDGMENT

REFERENCES

[1] R. Basnet, S. Mukkamala, and A. H. Sung, "Detection of phishing attacks: A machine learning approach," in *Soft Computing Applications in Industry*. Springer, 2008, pp. 373–383.

[2] S. Abu-Nimeh, D. Nappa, X. Wang, and S. Nair, "A comparison of machine learning techniques for phishing detection," in *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*. ACM, 2007, pp. 60–69.

[3] APWG. (2018) Phishing activity trends report, 3rd quarter 2018. [Online]. Available: https://www.apwg.org/resources/apwg-reports/

[4] R. M. Mohammad, F. Thabtah, and L. McCluskey, "Predicting phishing websites based on self-structuring neural network," *Neural Computing and Applications*, vol. 25, no. 2, pp. 443–458, 2014.

[5] A. K. Jain and B. B. Gupta, "A novel approach to protect against phishing attacks at client side using auto-updated white-list," *EURASIP Journal on Information Security*, vol. 2016, no. 1, p. 9, 2016.

[6] C. L. Tan, K. L. Chiew, K. Wong *et al.*, "Phishwho: Phishing webpage detection via identity keywords extraction and target domain name finder," *Decision Support Systems*, vol. 88, pp. 18–27, 2016.

[7] K. L. Chiew, E. H. Chang, W. K. Tiong *et al.*, "Utilisation of website logo for phishing detection," *Computers & Security*, vol. 54, pp. 16–26, 2015.

[8] I. Qabajeh, F. Thabtah, and F. Chiclana, "A recent review of conventional vs. automated cybersecurity anti-phishing techniques," *Computer Science Review*, vol. 29, pp. 44–55, 2018.

[9] R. S. Rao and A. R. Pais, "Detection of phishing websites using an efficient feature-based machine learning framework," *Neural Computing and Applications*, pp. 1–23, 2018.

[10] F. Feng, Q. Zhou, Z. Shen, X. Yang, L. Han, and J. Wang, "The application of a novel neural network in the detection of phishing websites," *Journal of Ambient Intelligence and Humanized Computing*, 2018.

[11] R. Mohammad, T. McCluskey, and F. A. Thabtah, "Predicting phishing websites using neural network trained with back-propagation." World Congress in Computer Science, Computer Engineering, and Applied Computing, 2013.

[12] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from urls," *Expert Systems with Applications*, vol. 117, pp. 345–357, 2019.

[13] M. Rajab, "An anti-phishing method based on feature analysis," in *Proceedings of the 2nd International Conference on Machine Learning and Soft Computing*. ACM, 2018, pp. 133–139.

[14] M. Babagoli, M. P. Aghababa, and V. Solouk, "Heuristic nonlinear regression strategy for detecting phishing websites," *Soft Computing*, pp. 1–13, 2018.

[15] Z. Pawlak, "Rough sets," *International journal of computer & information sciences*, vol. 11, no. 5, pp. 341–356, 1982.

[16] K. L. Chiew, E. H. Chang, C. L. Tan, J. Abdullah, and K. S. C. Yong, "Building standard offline anti-phishing dataset for benchmarking," *International Journal of Engineering & Technology*, vol. 7, no. 4.31, pp. 7–14, 2018.

[17] A. K. Jain and B. B. Gupta, "Towards detection of phishing websites on client-side using machine learning based approach," *Telecommunication Systems*, vol. 68, no. 4, pp. 687–700, 2018.

[18] D. Rodrigues, L. A. Pereira, R. Y. Nakamura, K. A. Costa, X.-S. Yang, A. N. Souza, and J. P. Papa, "A wrapper approach for feature selection based on bat algorithm and optimum-path forest," *Expert Systems with Applications*, vol. 41, no. 5, pp. 2250–2258, 2014.

[19] D. Dubois and H. Prade, "Rough fuzzy sets and fuzzy rough sets," *International Journal of General System*, vol. 17, no. 2-3, pp. 191–209, 1990.

[20] A. M. Radzikowska and E. E. Kerre, "A comparative study of fuzzy rough sets," *Fuzzy sets and systems*, vol. 126, no. 2, pp. 137–155, 2002.

[21] K. L. Chiew, C. L. Tan, K. Wong, K. S. Yong, and W. K. Tiong, "A new hybrid ensemble feature selection framework for machine learning-based phishing detection system," *Information Sciences*, 2019.

[22] N. Abdelhamid, A. Ayesh, and F. Thabtah, "Phishing detection based associative classification data mining," *Expert Systems with Applications*, vol. 41, no. 13, pp. 5948–5959, 2014.

[23] T. Peng, I. Harris, and Y. Sawa, "Detecting phishing attacks using natural language processing and machine learning," in *Semantic Computing (ICSC), 2018 IEEE 12th International Conference on*. IEEE, 2018, pp. 300–301.

[24] R. Sadeghi, T. Banerjee, and W. Romine, "Early hospital mortality prediction using vital signals," *Smart Health*, vol. 9-10, pp. 265–274, 2018.

[25] R. Sadeghi and J. Hamidzadeh, "Automatic support vector data description," *Soft Computing*, vol. 22, no. 1, pp. 147–158, 2018.

[26] M. Zabihimayvan, R. Sadeghi, H. N. Rude, and D. Doran, "A soft computing approach for benign and malicious web robot detection," *Expert Systems with Applications*, vol. 87, pp. 129–140, 2017.

[27] J. Hamidzadeh, M. Zabihimayvan, and R. Sadeghi, "Detection of web site visitors based on fuzzy rough sets," *Soft Computing*, vol. 22, no. 7, pp. 2175–2188, 2018.