Materials Today: Proceedings 47 (2021) 163–166

Contents lists available at ScienceDirect

Materials Today: Proceedings

journal homepage: www.elsevier.com/locate/matpr

Lexical features based malicious URL detection using machine learning techniques

Saleem Raja A. [a,*], Vinodini R. [b,*], Kavitha A. [c,*]

[a] Information Technology Department, University of Technology and Applied Sciences-Shinas, Sultanate of Oman, Oman
[b] Department of Computer Science, Mother Theresa Women's University, Kodaikanal, Tamil Nadu, India
[c] Department of Electronics and Communication Engineering, M.Kumarasamy College of Engineeringy, Karur, Tamil Nadu, India

ARTICLE INFO

Article history:
Received 19 March 2021
Accepted 4 April 2021
Available online 30 April 2021

Keywords:
Malicious URL detection
Machine learning
Feature extraction
Feature reduction

ABSTRACT

Most sophisticated cyber-attack technique used by the cyber criminals is creating and spreading malicious domain names or malicious URLs through email, messages, popups etc. Malicious URL are the web pages targeted towards the internet user to spread the malware, virus, warms etc once the user visited. Main intension of the attack is to steal the victim information, user credentials or install the malware in the victim's system. So, it is necessary to adapt the system which should detect the malicious URLs and prevent from the attack. Researchers suggest numerous methods but machine learning based detection method performs better then methods. This paper presents the light weighted method which includes only lexical features of the URL. The result shows the Random Forest classifier performs better than the other classifiers in terms of accuracy.

Selection and peer-review under responsibility of the scientific committee of the 12th National Conference on Recent Advancements in Biomedical Engineering.

1. Introduction

Increasing online web services makes the people life simpler than ever before. People are depending on online web services for their daily activities. This creates space for the cyber criminals to attack mass victims through specially designed, short living, malicious URL or web page. These malicious URLs are propagated through email, messages, twitter, facebook, popups, website ads etc. These URLs or websites contains some short of malicious files such as virus (ransomware) or malwares or keylogger etc. Once the victims click the malicious link, the malicious files are downloaded to the victim's system and opens the doors for the attack and data theft. Recent security reports states that more than 40,000 malicious URL are created every day, produces losses of $17,700 per minutes. More that 80% of the system around the world were compromised in 2020 [1,2]. Most of the (60%) of malicious URLs are spread through mail. So it becomes serious threat to the internet users. Rigid mitigation method is required to address the issue.

Common techniques used formalicious URL detection are blocklist based detection, heuristic rules-based detection and machine learning / deep learning-based detection [3]. Blacklist based technique uses huge list of URL which are already identified as malicious by programmatically or manually. Even though blacklist method gives low false positive and faster detection, poses some issues such as the database of blacklisted URL requires frequent and timely updates otherwise it will fail to detect the newly generated malicious URLs. Most of the cyber criminals never uses the existing malicious URLs for further attacks. To overcome these issues, researchers use heuristic rule-based detection technique which generalized rules are used for malicious URL detection. The rules the generated based on the features extracted from the datasets. In fact, feature extraction is the based for machine learning / deep learning technique. Problems in this technique are assigning weightage to the rules and fixing threshold value for each rule [4]. Threshold value may change based on datasets. On the other hand, machine learning technique extract the features from the dataset and train the model. The trained model is used for testing. Machine learning algorithms automatically assign the weightage of selected features and identify the malicious URL including newly generated malicious URLs.

Rest of the paper is organized as follows. Section 2 presents the existing research works in the research domain. Proposed system is presented in section 3. Section 4 presents the experiment results. Section 5 presents the conclusion and future work.

* Corresponding authors.
E-mail addresses: asaleemrajasec@gmail.com (A. Saleem Raja), avinodinimca@gmail.com (R. Vinodini), kavivenkat99@gmail.com (A. Kavitha).

https://doi.org/10.1016/j.matpr.2021.04.041
2214-7853/
Selection and peer-review under responsibility of the scientific committee of the 12th National Conference on Recent Advancements in Biomedical Engineering.

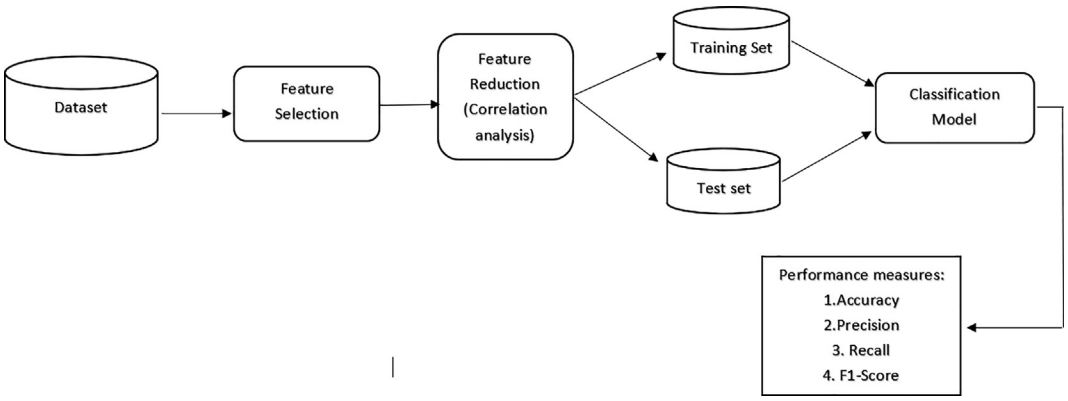

Fig. 1. Phases of proposed method.

Table 1
UNB dataset.

Category of URL	No. of URLs
Benign	35,378
Phishing	9965
Malware	11,566
Spam	11,942

2. Related work

Table 2
Lexical features.

Features	Features Category
IP_in_URL	Presence
URL_len	Length
Domain_len	Length
Dots_in_Domain	Count
Hyphens_in_Domain	Count
Underscores _ in_Domain	Count
Double-slashes _in_URL	Count
At(@)_in_URL	Count
Hash(#)_in_URL	Count
Semicolon(;)_in_URL	Count
And(&)_in_URL	Count
Http_in_URL	Count
Https_in_URL	Count
Numbers__in_URL	Count
Numbers _ratio _in_URL *	Ratio
Alphabets_in_URL	Count
Alphabet_ratio_in_URL *	Ratio
Lower_case_letters_in_URL	Count
Lower_case_ letters_in_URL *	Ratio
Upper_case_letters_in_URL	Count
Upper_case_ letters_ratio_in_URL *	Ratio
Special_char_in_URL	Count
Special_char_ratio_in_URL*	Ratio
English_words_in_URL*	Count
Random_words_in_URL*	Count
Avg_english_word_len_in_URL*	Length
Avg_random_words_in_URL*	Length

Continuous efforts of researchers, various new methods have been introduced to improve the detection accuracy and faster detection. Doyen et al. [3] presented a survey on malicious URL detection using machine learning. The paper highlights the pros and cons of different techniques in malicious URL detection in literature. Features used in machine learning techniques are grouped

Table 3
Confusion matrix.

	Classified Malicious	Classified Benign
Malicious URLs	True Positive (TP)	False Negative (FN)
benign URLs	False Positive (FP)	True Negative (TN)

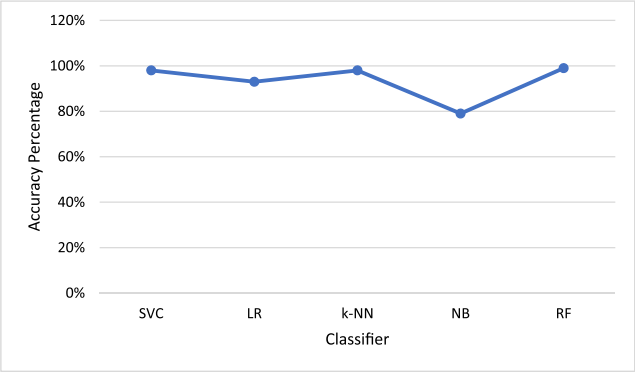

Fig. 2. Accuracy of different classifiers.

into four categories such as lexical features, host-based features, content-based features and other features (popularity features).

Dhamaraj et al. [5] proposed a multiclass classification method to detect the malicious URL. The paper considers 63 lexical features, 34 content related features and 18 hos-based features. Two algorithms were used for experiments such as SVM and multiclass CW learning. Data are collected from PhishTank, alexa and jwSpamSpy which includes 49,935 URLs (26041 benign URL, 8976 Phishing URL, 11,297 malware, 3621 spam URLs. 98.44% average prediction accuracy was achieved in the test.

Djaballah et al. [6] presented an approach to detect malicious URL in social network such as twitter. The paper includes 25 URL features (grouped under four categories 1. Lexical features 2. Webpage features 3. Host-based features and 4. Popularity features) and user account related features of twitter. Datasets includes UCI phishing datasets and 10,000 twitter accounts. Three machine learning algorithms (Logistic Regression, SVM and Random forest) were used for the experiments and it gives 90.28%, 93.43% and 95.51% accuracy respectively.

Ammara et al. [7] presented a method for phish website detection which uses different feature selection techniques and different machine learning techniques with PCA such as RF, NN, bagging, SVN, NB and KNN were used for experiments. The techniques are

Table 4
Result of classifiers.

Classifier	Ratio	Accuracy	Precision	Recall	F1-score	Execution time in sec
SVC	70: 30	0.98	0.99	0.98	0.98	251.66
LR	70: 30	0.93	0.94	0.93	0.94	2.696
k-NN (N = 5)	70: 30	0.98	0.98	0.98	0.98	2.233
NB (Gaussian)	70: 30	0.79	0.84	0.78	0.77	0.612
RF (100 trees)	70: 30	0.99	0.99	0.99	1.00	4.709

grouped as Stacking1 (RF + NN + Bagging) and Stacking2 (kNN + RF + Bagging) to improve the accuracy of classification. Stacking 1 produces highest accuracy of 97.4% than the other techniques.

Hafiz et al. [8] presented a way for selecting optimal features from dataset to detect malicious URL. UNB ang Kaggle datasets were used for feature extraction and testing. Nearly 41 word-based features, 36 count based features and 29 other features were used. KNN, SVM, LR, AdaBoost, Gradient Boost, ExtraTrees, RF and Voting classifiers algorithms were used for experiments. Feature's fitness and dependency with before target was tested before selecting the feature. The result shows 99% accuracy in UNB dataset and 94% accuracy in Kaggle dataset.

Ripon et al. [9] presents the experimental study on mostly used machine learning classifier such as SVM and Random forest. Lexical and host-based features are extracted from dataset and tested with ratio of 60:40, 70:30, and 80:20. And result confirms Random forest out performed than the SVM classifier. SVM accuracy result were more fluctuating than the Random forest.

Most of the existing work focus on the improvement of accuracy by selecting different feature sets. Host based, Content and Popularity based features incurs additional processing time and resources to extract the desired features [10]. Most of the works uses training and test ratio as 70% and 30% or 75% and 25% respectively.

3. Proposed method

The proposed method consists of following phases 1. Feature Extraction 2. Feature reduction 3. Train the Model 4. Testing as shown in the Fig. 1.

Feature extraction is the primary phase for machine learning techniques. UNB dataset 2016 is used for feature extraction and

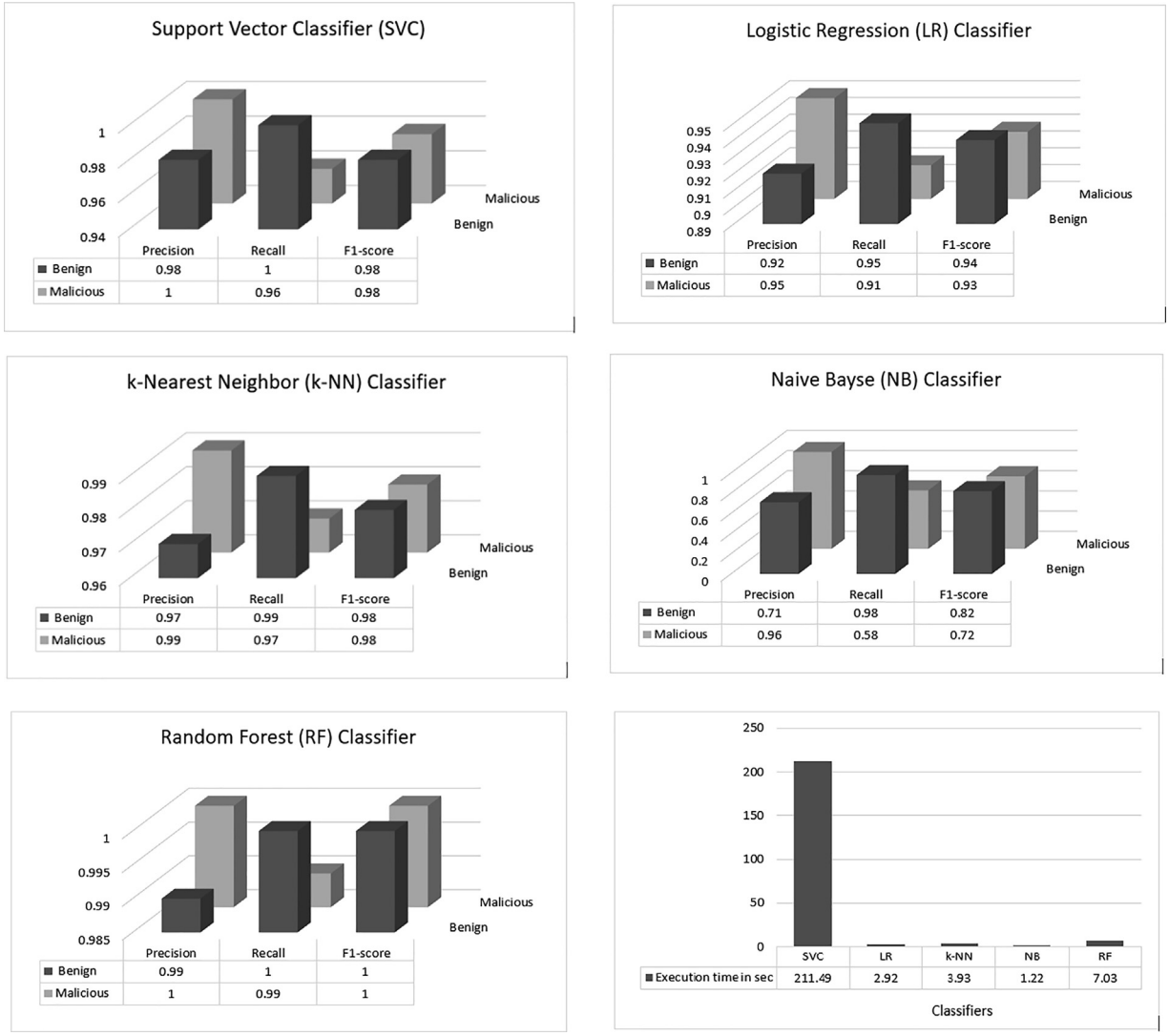

Fig. 3. Result of the performance metrics.

experiments with different machine learning algorithms. The UNB dataset includes benign, phishing, malware, spam url dataset as shown in Table 1.

Minimum set of optimal features which reduces the execution time and storage consumption. Initially identified 18 most common features and 9 newly identified features (*) in the dataset as shown in Table 2.

All extract features are not always suitable for classification. So it requires feature reduction method to identify the fitness of the features. Correlation analysis help to find the relationship between target feature and other features which in turn help to reduce the feature. Features are removed if there is no significant correction is present. Nearly 20 features are selected after correction analysis and 7 features are removed from features set (Underscores _ in_Domain, Double-slashes _in_URL, At(@)_in_URL, Hash(#) _in_URL, Https_in_URL, Numbers__in_URL, Special_char_ratio_in_URL).

4. Experiment results

Testing is conducted in the system using windows 10 operating and with 1.80 GHz intel core i7 processor and 16 GB RAM. Python with scikit learn package is used for programming to test different classifiers accuracy. The accuracy is computed based on the equation (1) and Table 3. Accuracy is defined as the percentage of correct decisions among all testing samples.

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

Other performance metrics evaluated with the proposed system are precision, recall and F1-score using formulae 2, 3 and 4 respectively.

$$precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 - Score = \frac{2xprecisionxrecall}{precision + recall} \quad (4)$$

Data of selected 20 features stored as a CSV file and fed in to the classifiers for training and testing with ratio of 70:30. Using scikit-learn python library, the classifiers (Support Vector Classification (SVC), Logistic Regression (LR), K-Nearest Neighbors (k-NN), Naïve Bayes (NB), Random Forest (RF)) are trained and tested. Accuracy of each classifier is tested, and results are presented in Table 4 and Fig. 2.

Performance metrics of different classifiers in presented in Table 4. The result shows the RF is outperformed than the other classifiers. But considering execution time and accuracy then k-NN gives better result than other classifiers as shown in Fig. 3.

5. Conclusion & future work

This paper presents the new method for malicious URL detection with fewer number of features extracted only from URL. That reduces execution time and storage requirements. It also highlights the recent research work in the domain and issues in the existing work. Result shows that random forest classifier is outperformed than the other classifiers. But considering execution time and accuracy then k-NN gives better result than other classifiers. In future, the research work will be extended with newer feature set and reduction techniques.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] https://www.csoonline.com/article/3153707/top-cybersecurity-facts-figures-and-statistics.html
[2] https://cyber-edge.com/wp-content/uploads/2020/03/CyberEdge-2020-CDR-Report-v1.0.pdf
[3] Doyen Sahoo, Chencho Liu, Steven Hoi, Malicious URL detection using machine learning: a survey, arXiv, 1(1) (2019).
[4] HarshalTupsamudre, Ajeet Kumar Singh, Sachin Lodha, Everything is in the name – a URL based approach for phishing detection, in: International Symposium on Cyber Security Cryptography and Machine Learning. Lecture Notes in Computer Science, vol. 11527, Springer, Cham, 2019, pp. 231–248.
[5] D.P.J. Patil, Feature-based malicious URL and attack type detection using multi-class classification, ISC Int. J. Inform. Security 10 (2) (2018) 141–162.
[6] B. Djaballah, B. Ghalem, A new approach for detection and analysis of phishing in social network: a case of twitter, in: Seventh International Conference on Social Networks Analysis, Management and Security (SNAM), 2020, pp. 1–8, https://doi.org/10.1109/SNAMS52053.2020.9336572.
[7] Ammara Zamir, Hikmat Ullah Khan, Tassawar Iqbal, Nazish Yousaf, Farah Aslam, Almas Anjum, Maryam Hamdani, Phishing web site detection using diverse machine learning algorithms.
[8] Hafiz Mohammd Junaid Khan, Quamar Niyaz, Vijay K. Devabhaktuni, Site Guo, Umair Shaikh, Identifying generic features for malicious URL detection system, in: IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2019, pp. 0347–0352. doi: 10.1109/UEMCON47517.2019.8992930.
[9] Katari Patgiri, Sharma Kumar, Empirical study on malicious URL detection using machine learning, in: International Conference on Distributed Computing and Internet Technology. Lecture Notes in Computer Science, vol. 11319, Springer, Cham. https://doi.org/10.1007/978-3-030-05366-6_31.
[10] Bc. Andreaturiokova, Detecting malicious URLs, Master thesis, Masaryk University, 2019.