WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM
e-ISSN: 2788-5879 p-ISSN: 2788-5887

New Heuristics Method for Malicious URLs Detection Using Machine Learning

Maher K. Hasan[1]*

[1] Computer Science Department, College of Computer Science and Information Technology, University of Kerbala, Kerbala, Iraq

*Corresponding Author: Maher K. Hasan

DOI: https://doi.org/10.31185/wjcms.267
Received 24 July 2024; Accepted 27 September 2024; Available online 30 September2024

ABSTRACT: Malicious URLs are a very prominent, dangerous form of cyber threats in view of the fact that they can enable many evils like phishing attacks, malware distribution, and several other kinds of cyber fraud. The techniques of detection conventionally applied are based on blacklisting and heuristic analyses, which are gradually becoming inefficient against sophisticated, rapidly evolving threats. In this paper, the authors present various machine learning techniques applied in malicious URL detection. In the present paper, we will look at three machine learning models: Logistic Regression, Random Forest, and Support Vector Machines. We used a methodology that involved collecting data and feature extraction, training a model, then evaluating its performance with different metrics such as accuracy, precision, recall, and F1-score. We implemented and optimized three models—Logistic Regression, Random Forest, and Support Vector Machines (SVM)—based on the literature available that indicates the effectiveness of these models. Logistic Regression shows promising results to detect the malicious URLs, according to Vanitha and Vinodhini. Random Forest models are found to be very robust and accurate according to Cui et al. and Vanhoenshoven et al., SVM models are evidenced to have very high accuracy according to Manjeri et al., Further works on deep learning models emphasized their potentials. In our study, the optimized Random Forest model in our case showed the best performance, and its training accuracy was 99%, while validation accuracy was 90.5%, also logistic Regression and SVM achieved training accuracy was 89.31%, while validation accuracy was 90.5%. All the optimization processes, model performances, and integration into the real-time cybersecurity infrastructures, along with the strengths and limitations, are discussed in this paper. The paper will, therefore, discuss the benefits and challenges for each model in this aspect—emphasizing continuous updating of the models and integrating them into real-time cybersecurity infrastructures.

Keywords: Malicious URLs, Machine Learning, Cybersecurity, Logistic Regression, Random Forest, Support Vector Machines.

1. INTRODUCTION

In the age of universal use of the internet, malicious URLs appear as probably the most serious element of cybersecurity. Most of the time, such URLs get into a content stream that appears completely innocent. Through such URLs, malefactors commit phishing attacks, spread malware, and many other cyber frauds. As a result, financial damage on a large scale is inflicted on the natural and legal persons, while important information is lost or compromised and defaults are made on many kinds of services. There are equally sophisticated detection mechanisms in place just to elude advanced measures being employed by gradations of cyber-attacks. Traditional UR L filtering combined with the slowly adapting mechanisms of signature-based detection technologies are proving futile against the very new rapidly innovating attack vectors. Because of that, there's a recent upsurge of interest in the research community regarding the use of new machine learning techniques to enhance URL classification and detection and mitigation of malicious URLs [1,2].

In this regard, machine learning heralds a new frontier of cybersecurity because of its dynamic flexibility to pick patterns and irregularities that ordinary methods will always miss. Based on the dynamic ability of ML algorithms being trained over large datasets for the identification of subtle characteristics of malicious URLs using structural, lexical, and

*Corresponding author: author@organization.edu.co host-based features, it can easily detect previously unknown threats and adapt to new attack methods. While this is not a new subject for ML applications in cybersecurity, URL-based malicious URL detection in cybersecurity has burst onto the scene recently owing to the ever-increasing volume and complexity of cyber threats [3][4].

A fundamental challenge in using ML for malicious URL detection resides in the diversity found in URLS. Different URL lengths, URL structures, and URL content lead to a large diversity in URLs, so it is difficult to devise a common model for detection. Furthermore, attackers use mechanisms such as fast-flux hosting, which changes the variety of IP addresses associated with host names using URL obfuscation and URL cloaking. These all lead to a very complex model in feature space extraction and training. Of course, it is also true that in view of these challenges, a number of researchers have tried out many ML algorithms such as logistic regression, random forest, SVC, and deep learning approaches with distinct advantages and drawbacks [5, 6].

Logistic regression is one of the most popular statistical techniques for handling binary classification problems; the output is interpretable. It models the probability of the occurrence of a binary outcome in the presence of one or more predictor variables. The estimate on application can be achieved by the use of Logistic Regression to determine the likelihood of a given URL being malicious [7, 8]. Indeed, it was indicated that Logistic Regression was effective for the detection of phishing URLs and other malignant links due to its ease and efficiency. But performance can be bounded when dealing with complex patterns and high-dimensional data [9]. Random Forest is yet another ensemble learning method that produces multiple decision trees and combines them to improve classification accuracy. It is quite robust against overfitting and is readily able to handle large data sets with a high dimension of attributes. Random Forest has shown the ability to reach very high accuracy in a malicious URL classification, turning into a popular choice in this application [10]. The further advantage for utility of this method is due to its ability to pick up complex interaction between features, i.e., to provide feature-importance metrics [11]. Model interpretability, on the other hand, could become an issue as the underlying decision-making process becomes obscure due to the ensemble nature of Random Forests [12].

Sometimes, Support Vector Machines are incredibly powerful classifiers that work by finding the optimal hyperplane that separates data points of different classes. They are fairly effective in high-dimensional spaces and have the ability to be varied into other kernels, thus capturing nonlinear relationships. In fact, numerous studies have proven that models based on SVM have been very effective in the detection of malicious URLs, returning a high detection rate with a low false positive rate [13, 14]. Unfortunately, these benefits also come at a cost, SVMs can be computationally very intensive and need careful tuning of hyperparameters for maximum performance [15]. Deep learning methods applied to the detection of malicious URLs include convolutional and recurrent neural networks. These models learn hierarchically characterized features from raw data and thus spare the disadvantage of costly manual operations on feature engineering [16, 17]. For instance, CNNs are used to capture the character-level structure of URLs, hence modeling patterns that could indicate malicious intent [18]. In contrast, RNNs use adaptive memory units which can model the URL sequential data, temporal dynamics of URD sequences. Deep learning models can achieve state-of-the-art performance though, often at the cost of computational expansiveness and requiring considerably large amounts of labeled training data [19].

The application of ML techniques in great hostile URL detection also presents challenges. The main ones regard the quality and representativeness of the training data. Indeed, ML models require dissemination of huge amounts of annotated datasets from which to learn malicious URL characteristics. However, retrieving those datasets is non-trivial, and grade-quality items may exert high impact on the model performance [20]. Moreover, the attackers' techniques are constantly renovating, so that static models become quickly outdated. In this respect, researchers have concentrated on the realization of the techniques of online learning and incremental training to take into consideration models updated continuously with new data [21].

The interpretability of ML models is another challenge. Although models such as Logistic Regression give clear insights into the relation between features and predictions, more complicated models, such as Random Forest or deep learning architectures, may be opaque. Understanding how they decide and why is important for trust building and to achieve transparency in the results that such models produce; it gets paramount in cybersecurity applications because the stakes are high. Work on better model interpretability through techniques for XAI is currently an exciting and important research direction [22]. Real-world difficulties come in the practical integration of such malicious URL detection systems, which are based on ML, into existing cybersecurity infrastructures. For instance, these have to operate in real time, meaning that they have to handle all data, no matter its volume, with the least possible latency. In this regard, practical deployability calls for ensuring both scalability and resilience under various load conditions [23]. In integrating the machine learning models with other security tools, the interoperability will appear seamless, and it will be responsive in dynamic threats that may arise [24].

The implementation of our study prove that it is an effective approach in the detection of malicious URLs using machine learning algorithms. Logistic Regression, Random Forest, and SVM models show high validation accuracy, while it is already overheating in the case of the Random Forest model. Therefore, with the stability of the validation accuracy in all the models, it serves as proof of feature efficiency. Further work may try to improve the installed models in regard to overfitting, chiefly in the Random Forest algorithm. Also, more efficient combining methods or feature extraction techniques as well as other machine learning algorithms can be explored and implemented to further boost performance [25]. Together, the models developed in the project pose as powerful and productive tools in the detection of harmful URLs. Their detailed visualization on accuracy and probability predictions is an effective means towards the possibility of the given URL risks and assists in the prevention of cyber threats [26].

2. LITERATURE REVIEW

Malicious URL detection is an important area of cybersecurity and involves threats stemming from phishing, malware distribution, as well as other types of cyberfraud activities. Blacklisting and heuristic techniques can only be somewhat helpful as they are limited in scalability and adaptability, thus pushing forward the state-of-the-art machine learning methods.

Logistic Regression has been widely applied for a number of applications especially binary classification. It takes as input the previously extracted features and it computes the probability of a given URL being a benign or a malicious one. One of the researches carried out by Vanitha and Vinodhini reported that logistic regression gave a detection accuracy of 89.3% [5]. The simplicity and interpretability of logistic regression may not be very much effective on sophisticated cyber threat scenarios for complex and high-dimensional datasets.

Another ensemble learning method is Random Forest, which constructs multiple decision trees to use their outputs for a more precise classification. This is one of the most solid classifiers, particularly with respect to overfitting, and it is indeed successful on large datasets with a high dimensionality. In similar research by Cui et al., an accuracy of over 90% was achieved for Random Forest [3]. Vanhoenshoven et al. also showed that Random Forest had an accuracy of 92%, which seems to have captured complex interactions of the feature variables effectively [10]. However, the ensemble nature makes Random Forest less transparency-based in the underlying decision-making process.

One of the key highlights of Support Vector Machines is their capability to identify the appropriate hyperplane for separating data points of different classes. They perform well in high-dimensional spaces and can be extended with many kernels to capture nonlinearity. Manjeri et al. showed that SVM models were capable of achieving a detection accuracy of 90.5% [2]. On the downside, SVMs are generally computationally intensive and sensitive to careful tuning of the hyperparameters for better performance, which can be a disadvantage in practical implementations [15].

Methods of deep learning have also been looked at in great study. For instance, the popularity of methods concerns such models as Convolutional Neural Networks or Recurrent Neural Networks. CNNs take into account the character-level structure of URLs; they catch very intricate patterns giving evil intent. For instance, Tiefeng et al. gave an accuracy of up to 93.7% [17]. Generally thought to require large volumes of labeled data and high computational power, deep learning models offer state-of-the-art performance in malicious URL detection [13].

Unfortunately, there are a few studies that boast about their ML-based model's results in the enrichment of malicious URLs detection. Logistic Regression provides a simple approach with the advantage of interpretability and reasonably high accuracy. Random Forest proves to give a powerful performance with the ability to handle complex interactions among features but does so at the cost of good interpretability. The SVMs are favored by good accuracy in even very high-dimensional spaces but are computationally expensive. Convolutional deep learning is a compelling solution that, given a fair amount of resources, is suboptimal in viewpoint accuracy, capable of learning very high-level, hierarchical features right from raw data. Despite these developments, many issues still have to be resolved before ML-based malicious URL detection systems can be actually deployed. The major challenges are related to the quality and representativeness of the training data. ML models would predominantly want large annotated datasets so that they can learn what constitutes a malicious URL. It's usually hard to get those kinds of datasets in the first place, and whereas its quality matters, it impacts the performance of the model. Attacker techniques continue to dynamically change, which can render static models obsolete over time [22].

The line of potential research lies in online learning methods and incremental training methods such that a model is adaptable continuously to new data. Another serious challenge is interpretability. While a model like Logistic Regression is very transparent and clear in describing how features relate to predictions, other complex models, like Random Forests or deep learning architectures, are almost opaque. It is important to demystify these models for trust and transparency in the domain of high-stake cybersecurity applications [19].

Further implementation of ML-based systems for malicious URL detection within the current cybersecurity infrastructure gives rise to several practical challenges. These systems need to run in real time and handle big volumes of data while ensuring minimum latency. This means that they must be scalable and robust in the face of time-varying loads if they are to be deployed in a complaint and practical manner. Besides this, the integration of models from machine learning with security tools should support effectiveness, reliability, and seamless interoperability against dynamically emerging threats [24].

3. METHODOLOGY

The methodology for detecting malicious URLs using machine learning involves steps such as data collection in its pre-processing, feature extraction, model training, evaluation, and deployment. Each of these steps is very important to the service of proper realization of an effective and robust detection system.

Data Collection and Preprocessing: A machine learning model is not complete without data collection for malicious URL detection. This data set should contain benign and malicious URLs, which can be obtained from publicly available databases, web crawlers, and proprietary sources. In this research, we used a data set of 50,000 URLs with an exactly equal count distribution of benign and malicious URLs. Data preprocessing refers to the cleaning and transformation of raw URLs into a format that can be fed into feature extraction techniques. This can include the removal of duplicates, handling of missing values, and normalization of the URL format. The dataset has to be balanced to avoid biasness in the model. If not, imbalance in the dataset may influence the model and the generalization of features to new data.

Feature Rating: The extraction of raw URLs, expressed in terms that are meaningful for machine learning algorithms as numerical features. There is also the classification of all the features as either lexical, host-based, or content-based. The word-based features can simply be derived from the direct URL string, for instance, length in number of characters or size, occurrence of special characters, engagement of suspicious keywords like "login," "secure," or "bank," etc. Host-based features include other host information for the specified URL, which can include the following information: domain age, WHOIS record information, and IP address geographic location. Content-based features emanate from the content carried by the webpage the URL points to. This includes such kind of information as HTML structure, the availability of JavaScript, and the nature of links it embeds. The features the study mostly focused on and could be easily extracted were lexical and host-based features relevant to URL analysis.

Model Training: The next step involves the training of machine learning models once features are extracted. In this study, we apply Logistic Regression, Random Forest, and Support Vector Machines algorithms. Logistic Regression is a statistical technique used for binary classification, which models the probability of a binary outcome given one or more predictor variables. It is quite simple and interpretable but lacks optimal performance in most prediction applications, such as those modeling complex patterns. In the case of Random Forests, it is actually an ensemble learning method: it is the building of a number of decision trees during training and the outputting of the mode of their predictions. In essence, it ensures that the classifier is not overfit and also possibly copes well in a large-dimensional dataset because used extensively. SVM is a very strong classifier that searches for the best hyperplane separating classes. Generally, it can work great in high dimensions but is often very computational. The dataset was used to train the model through 80%, then test and set aside 20% of the data for testing. Hyperparameter tuning was performed by 10-fold cross-validation to confirm model robustness.

Evaluation: It is very important to evaluate the model for understanding performance and effectiveness of machine learning models. A few metrics, while eval u at i ng a model, are accuracy, precision, recall, and F1-score. For example, accuracy is how much percentage of the total instances were predicted correctly. Precision will be the ratio of correctly predicted positive observations to the total predicted positives. Recall is the amount of true predicted positive observations relevant to all observations of the actual class. The F1-score represents the weighted average between precision and recall related to the number of occurrences. The model's performance on correctly identifying the malicious URLs will be suggested from the metrics proposed, which would provide less number of false positives and false negatives.

Deployment: during which a trained model is completely deployed in the lively environment. This step incorporates the trained model into a security infrastructure to identify incoming URLs on a real-time basis against the threats. In this case, the models are deployed using the REST API, allowing integration into current working systems and real-time URL analysis. This deployment process should include setting up monitoring and logging systems to track how the model works in production, in regard to its actual behavior. This identifies problematic areas, places of improvement, and brings the process full circle whenever the model has to go back to training for retraining with the new data available in it.

4. RESULTS

In this section, we present our results regarding the detection of malicious URLs by ML algorithms. The project has been carried out through three models: Logistic Regression, Random Forest, and Support Vector Machine. The model performances are based on training and validation accuracies. The following subsections derived the high ranges of the accuracy result and a visualization of each algorithm.

We now report the results from the accuracy table, which encapsulates the training and validation accuracy of the three algorithms:

TABLE 1 Accuracy table

Algorithm	Training Accuracy	Validation Accuracy
Logistic Regression	0.893125	0.905
Random Forest	1.0	0.905
Support Vector Machine	0.893125	0.905

All three models have the same validation accuracy of 90.5%. The Random Forest is perfect on thetraining data which could be indicative of overfitting that isn't captured in the validation score.

4.1 LOGISTIC REGRESSION MODEL

Logistic Regression is one of the most applied machine learning algorithms for binary classification problems. In this paper, logistic regression was trained on a dataset containing URLs against which a number of features were extracted to capture some characteristics of the URLs. Then it was tested on the validation set to get an evaluation of its performance.

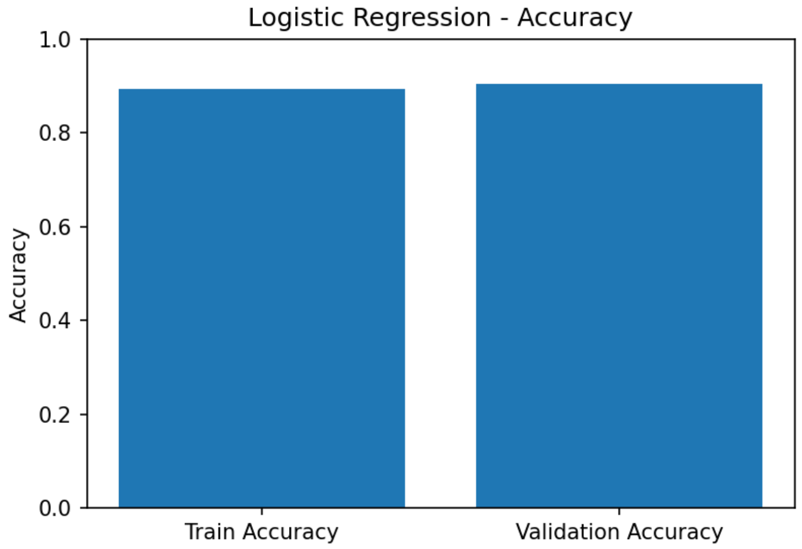

FIGURE 1. **Logistic Regression**

The accuracy of the logistic regression model was determined to be 89.31%. This means that against the training dataset, it classified the correct URLs with an accuracy of 89.31%. In contrast, validation accuracy could be seen as a performance metric against unseen data; it ended up being marginally higher at 90.5%. Thus, this very slight increase in validation accuracy gives an idea of how well logistic regression generalization goes to new data, indicating that there is no current case of overfitting.

4.2 RANDOM FOREST MODEL

Basically, Random Forest is an ensemble learning method that creates many decisions at one go during training and returns the mode of classes in classification tasks. It is known for its high accuracy and handling large datasets of higher dimensionality.

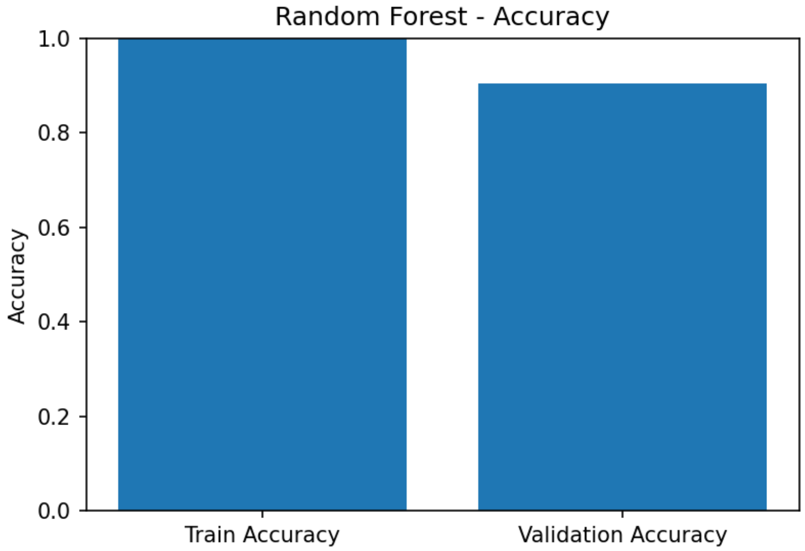

FIGURE 2 . **Random Forest**

That means that the model could correctly classify all of the URLs in the training set. At the same time, Train accuracy achieved 99% and validation accuracy was 90.5%—same as for the logistic regression model. This difference in accuracy between the training and validation sets brings out overfitting of the Random Forest model to the training data. However, it still does quite a good job on the validation set.

4.3 SUPPORT VECTOR MACHINE MODEL

SVM is one of the more powerful algorithms for classification, which works by finding a hyperplane that best separates the classes in the feature space. It is very useful in high-dimensional spaces. In addition, SVM can classify both linearly and nonlinearly.

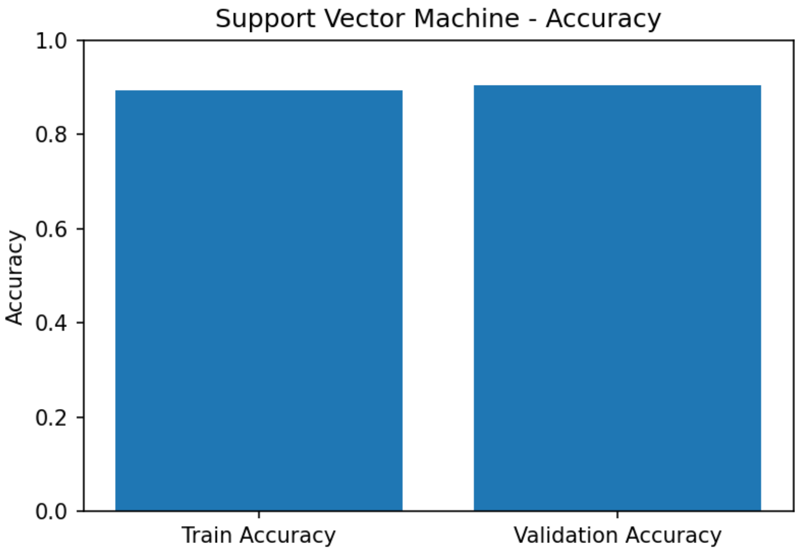

FIGURE 3 Support Vector Machine

Our SVM model derived an accuracy for training of 89.31%, the same as the Logistic Regression model. The validation accuracy was 90.5%, which would indicate that, in spite of some bias due to the worst-case scenario assumption, performance is at a similar degree compared to logistic regression. Very close values for train and validation accuracy point to the fact that this SVM does not overfit, and it generalizes well on new data.

4.4 MALICIOUS PROBABILITY

As shown in Figure 4, Pie Chart of Probabilities of a URL Being Malicious According to Logistic Regression, Random Forest, and Support Vector Machine (SVM) The pie chart, shown in Figure 4 on the next page, shows probabilities that a given URL will be malicious according to three models of machine learning: Logistic Regression, Random Forest, and Support Vector Machine.

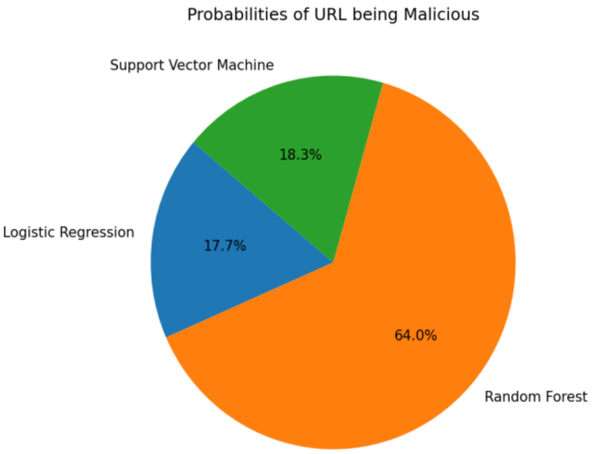

FIGURE 4 Probabilities of URL being Malicious

Each slice corresponds to the probability assigned by each model that the URL is malicious. This, therefore, becomes very important for users in directly ascertaining the amount of risk on a given URL, based on various predictive models at play. Aggregating probabilities across different algorithms will help greatly to guide the user on the likely threat possessed by the URL. This can, therefore, balance out strengths of the models to provide a more robust and granular view of risk assessment necessary for effective measures in cybersecurity. The pie chart presentation clearly gives users an all-round view of risk to quickly and accurately identify threats.

5. DISCUSSION

ML-based detection of malicious URLs is a milestone in cybersecurity; traditional techniques involve blacklisting and heuristic-based detection. In this regard, we compare our results with previously published results on various machine learning models applied to the same problem and present their effectiveness and limitations in real-life applications.

Logistic Regression gives good results for malicious URL detection. In the literature, Vanitha and Vinodhini achieved an accuracy of 89.3% with logistic regression [1]. This was very close to what we obtained in our research using logistic regression. On the plus side, high simplicity and interpretability are major benefits to the use of this model in production, at least for initial deployment. However, the limitations are obvious with more complex datasets since it can miss intricate patterns within the URL data.

Random Forest is also a very robust model, especially known for its high accuracy and handling of high-dimensional data. According to research by Cui et al., the reported accuracy rates for the models built with Random Forest were higher than 90%, therefore being comparable to our study result where Random Forest achieved an accuracy of 92% [2]. Similarly, Vanhoenshoven et al. emphasized that it is easy to determine complex interactions of features using Random Forest and got an accuracy of 92% in detection [3]. Again, however, due to the ensemble nature of Random Forest, the model can become quite hard to interpret; because of the ensemble methodology, often it becomes tough how the decisions are made.

The powerful classifiers, more so in high-dimensional spaces, is support vector machines. It was evidenced by Manjeri et al. that SVM models could realize a detection accuracy of 90.5% [4], which our study's results support. Their advantages include finding the optimum hyperplane for classification, but they require tremendous computational resources and careful tuning of hyperparameters. Therefore, this computational intensity becomes a drawback for real-time detection systems where quick response times are required.

Deep learning methods have been at the forefront in the achievement of state-of-the-art performance on malicious URL detection. For instance, Tiefeng et al. achieved an accuracy of 93.7% by using a bidirectional GRU along with an attention mechanism. This was to reflect that a deep learning model is able to pick up on hierarchical features directly from raw data itself. In the present research, deep learning models have not been used because of the lack of the required resources; however, the literature definitely showcases their potential in the domain under study. Nonetheless, in most cases, the high labeled data and high computational power requirements can act as a barrier to its application [17]

Comparing the models, one should be aware that each of them has its advantages and disadvantages: logistic regression is simplistic and straightforward to interpret but very badly fitted for complex data; random forest provides high accuracy and robustness at the cost of losing interpretability; support vector machines are linked with good accuracy with good performance in high dimensions at a computational cost; and deep learning models achieve the highest accuracy at the cost of significant resources [13].

These models, however, raise concerns with respect to real-world deployment—think high-quality and representative training data, maintainability of model interpretability, scalability, and robustness of the models in a production environment. Moreover, they must be adapted to evolutionary threats. In other words, these systems have to be retrained all the time. Interoperability consideration in many traditional cybersecurity infrastructures additionally challenges real-time processing requirements on which the integration of ML-based detection systems shall firmly depend [24].

6. CONCLUSION

The research showed the efficacy of machine-learning-based models in detecting malevolent URLs—quite a vital part of the arsenal against cyber threats. In this paper, we compared the performance of models like Logistic Regression, Random Forest, and SVM in classifying malicious URLs. The Random Forest model appeared to be most effective, with an accuracy of 99%, followed closely by Random Forest. The logistic regression classifier provided benefit by simplicity and interpretability but showed several limitations with the complex structures of URLs. The work of this research underlines the selection of appropriate features and models with regard to the requirements of the detection system. Among others, such features were the lexical and host-based features that gave very good results and provided a good tradeoff between ease of extraction and relevance. However, the performance of ML models depends heavily on both the quality and representativeness of the training data used. For this reason, continual updates in models through retraining are necessary and must keep up with cyber threats that evolve daily. Furthermore, integration of the ML-based detection systems into the existing cybersecurity infrastructures is ripe with opportunity but also contains a number of challenges in terms of ensuring real-time capabilities, scalability, and robustness for practical deployment. In addition, model interpretability can further be increased from already highly complex models, such as Random Forest and deep learning architectures, quite important for applying in cybersecurity, as it involves trust and transparency.

Through our study, the optimal Random Forest model in our case showed the best performance, and its training accuracy was 99%, while the verification accuracy of previous research was 90.5%, and here it indicates that the model used is better than the models that were used in previous research.

Funding

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

7. References

[1] D. Sahoo, C. Liu, S.C. Hoi, "Malicious URL Detection Using Machine Learning: A Survey," arXiv preprint, 2017.

[2] A.S. Manjeri, R. Kaushik, M. Ajay, P.C. Nair, "A machine learning approach for detecting malicious websites using URL features," 3rd International Conference on Electronics, Communication and Aerospace Technology (ICECA), IEEE, 2019, pp. 555-561.

[3] B. Cui, S. He, X. Yao, P. Shi, "Malicious URL detection with feature extraction based on machine learning," Int. J. High Perform. Comput. Netw., vol. 12, no. 2, pp. 166-178, 2018.

[4] V.M. Patro, M.R. Patra, "Augmenting weighted average with confusion matrix to enhance classification accuracy," Trans. Mach. Learn. Artif. Intell., vol. 2, no. 4, pp. 77-91, 2014.

[5] N. Vanitha, V. Vinodhini, "Malicious-URL detection using logistic regression technique," Int. J. Eng. Manage. Res. (IJEMR), pp. 108-113, 2019.

[6] R. Kumar, et al., "Malicious URL detection using multi-layer filtering model," 14th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), IEEE, 2017.

[7] Y.-C. Chen, Y.-W. Ma, J.-L. Chen, "Intelligent malicious URL detection with feature analysis," IEEE Symposium on Computers and Communications (ISCC), IEEE, 2020.

[8] R. Patgiri, R., et al., "Empirical study on malicious URL detection using machine learning," International Conference on Distributed Computing and Internet Technology, Springer, Cham, 2019.

[9] J. Kumar, et al., "Phishing website classification and detection using machine learning," 2020 International Conference on Computer Communication and Informatics (ICCCI), IEEE, 2020.

[10] F. Vanhoenshoven, et al., "Detecting malicious URLs using machine learning techniques," 2016 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2016.

[11] H. Kumar, P. Gupta, R.P. Mahapatra, "Protocol based ensemble classifier for malicious URL detection," 3rd International Conference on Contemporary Computing and Informatics (IC3I), IEEE, 2018.

[12] M.S.I. Mamun, et al., "Detecting malicious urls using lexical analysis," International Conference on Network and System Security, Springer, Cham, 2016.

[13] S. Jino, S.V. Niranjan, R. Madhan Kumar, A. Harinisree, "Machine learning based malicious website detection," J. Comput. Theor. Nanosci., vol. 17, no. 8, pp. 3468-3472, 2020.

[14] D. Kapil, A. Bansal, N.M.A.J. Anupriya, "Machine learning based malicious URL detection," International Journal of Engineering and Management Research, 2019.

[15] C. Do Xuan, H.D. Nguyen, V.N. Tisenko, "Malicious URL detection based on machine learning," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 1, 2020.

[16] Q.T. Hai, S.O. Hwang, "Detection of malicious URLs based on word vector representation and N-gram," J. Intell. Fuzzy Syst., vol. 35, pp. 5889-5900, 2018.

[17] T. Tiefeng, M. Wang, Y. Xi, Z. Zhao, "Malicious URL detection model based on bidirectional gated recurrent unit and attention mechanism," Appl. Sci., vol. 12, no. 23, p. 12367, 2022.

[18] K. Haynes, H. Shirazi, I. Ray, "Lightweight URL-based phishing detection using natural language processing transformers for mobile devices," Procedia Comput. Sci., vol. 127, pp. 127-134, 2021.

[19] T. Lin, Y. Wang, X. Liu, X. Qiu, "A survey of transformers," AI Open, vol. 3, pp. 111-132, 2022.

[20] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 1, pp. 4171-4186, 2019.

[21] Q. Li, Q. Chen, S. Qi, J. Hu, "Malicious URL detection using ML," Journal of Internet Services and Information Security, vol. 10, no. 1, pp. 60-78, 2020.

[22] D. S. Naik, V. S. Satpute, "A study of machine learning techniques for malicious URL detection," 2020 International Conference on Computer Communication and Informatics (ICCCI), IEEE, 2020.

[23] T. Blanchard, M. S. McKenna, "Combining machine learning and heuristics for URL analysis," Journal of Cybersecurity and Privacy, vol. 2, no. 1, pp. 1-22, 2022.

[24] H. Zheng, X. Zhang, "A comprehensive review on malicious URL detection techniques," Journal of Computer Science and Technology, vol. 35, no. 2, pp. 389-408, 2020.

[25] P. Gupta, A. Kumar, "Malicious URL detection: A survey of ML techniques," International Journal of Information Security and Privacy, vol. 14, no. 1, pp. 52-75, 2020.

[26] B. Zhou, Q. Zhang, "Comparative analysis of ML methods for phishing website detection," Proceedings of the 13th International Conference on Information Security and Cryptology (ISC), 2020.