Evaluation on Malicious URL Detection with Different Features Based on Various Machine Learning Algorithms

Xiang Guo
Faculty of Engineering & IT, The University of Melbourne, Parkville VIC 3052, Australia

Keywords: Machine Learning; KNN; Neural Network; Logistic Regression; URL Detection.

Abstract: With the escalating demand for cybersecurity, the identification of malevolent Uniform Resource Locators (URLs) has assumed paramount significance in defending against cyber threats. Various techniques, ranging from blacklists and heuristics to machine learning methods, have been employed for the purpose of detecting malicious URLs. Among these methodologies, machine learning stands out prominently due to its scalability, adaptability to emerging threats, and capacity to uncover threats that were hitherto unknown. This paper focuses on analyzing deep learning learning methods to detect malicious URLs compared with two traditional machine learning algorithm: Logistic Regression (LR) and K Nearest Neighbour (KNN). Scratching and collecting over 200,000 data to train the model and make prediction and evaluation. The result shows that the deep learning algorithm could achieve much higher scores than the other two machine learning models, but has much lower efficiency. The KNN model has better performance on selected feature group than hybrid feature group. The LR model could achieve higher performance on huge dataset and extremely complex feature group.

1 INTRODUCTION

Over time, as the demand for cybersecurity continues to grow, ensuring the safety and security of individuals and organizations has become paramount. Malicious Uniform Resource Locators (URLs) can lead to various cybersecurity threats such as phishing attacks, malware distribution, or unauthorized access to sensitive data. Detection of malicious URLs often involves analyzing the URL structure, evaluating the domain reputation, and assessing the content hosted on the linked page. Thus, the numbers of malicious url detection approach have significantly increased. Overall, these approaches could be classified into three types: Blacklist Approach, Heuristic or Rule-based Approach, Machine Learning Approach (Sahoo D et al 2017). The machine learning approach offered benefits in terms of scalability, adaptability to changing threat landscapes, and the ability to detect previously unseen or zero-day threats, which make machine learning approach quite outstanding compared with the other two approaches. Therefore, the research and exploration of novel machine learning methods to detect malicious url has become increasingly important.

Three main feature groups have been identified of an URL: Lexical group, host-based feature group, correlated feature group (Do et al 2020), numbers of novel machine learning algorithm and ideas has been posted for different features and types of URL to train and predict. In the study of B. B. et al in 2021, Gupta et al. have proposed an approach to extract and pre-process the feature vectors based on lexical feature group. This approach only uses nine features based on the lexical properties of URLs to reach 99.57% accuracy. For the group of host-based features, it is usually referred to malicious domains which hidden in URLs (Palaniappan et al 2020), the paper identified several common the Domain Name System (DNS-based) features of a domain name, such as Autonomous System Number and IP addresses to explore an active DNS analysis approach and trained a logistic regression classifier to get around 60% accuracy. Except for single group of features, researching hybrid correlated features group is also worthwhile. One hybrid feature selection algorithm posted by Kumar's study in 2023, it converts the text features into numerical vectors using Word2Vec and apply Principal Component Analysis (PCA) feature selection algorithm to reduce the dimensionality combined with Natural Language Processing (NLP)

Guo, X.
Evaluation on Malicious URL Detection with Different Features Based on Various Machine Learning Algorithms.
DOI: 10.5220/0012808400003885

In *Proceedings of the 1st International Conference on Data Analysis and Machine Learning (DAML 2023)*, pages 338-342
ISBN: 978-989-758-705-4

features to train and test. The result demonstrated that hybrid Features generally outperformed Word Vectors across all algorithms, thus, it could improve the overall accuracy.

However, most of these kinds of research are lack of assessment and analysis of efficiency, robustness, usability and performance. Furthermore, a significant portion of these studies relies on outdated datasets, primarily from around 2016. Moreover, a subset of these studies continues to employ basic linear regression for model training, suggesting ample room for more advanced exploration and research. The main contribution of this paper is to reproduce some of detection process but replace them with neural-network machine learning architecture or K nearest neighbours to explore the potential extension and limitation as well as make assessment on performance by comparing with previous work.

2 METHODS

2.1 Dataset

2.1.1 Dataset Preparation

In this research, open-source URL dataset (ISCX-URL2016) was collected as main resources. The ISCX-URL involves types of maliouсs URLS including Benign, Spam, Phishing, Malware and Defacement URLs. The amount of URLs is over 200,000 in this dataset, which provide significantly common and convenient resources to train the model and evaluate the result in all kinds of situation.

2.1.2 Pre-Processing Values

Several steps need to be done to pre-process the dataset: 1) Gathering and sorting the dataset by length to make the dataset straightforward to check and analyze. 2) Getting rid of the duplicated data since they will make the model overfitting. 3) Seperating the URLs into words and vectorized the words. In this research, using Word2Vec and Tfidf to vectorize the URLs, which make it easy to fetch and select features of URLs. 4) Deleting the suffix of URLs, such as com, cn, au. Because all kinds of these suffixes are not key words of training the model. 5) Around 200, 000 data has been collected after the pre-processing and is used in this research. The whole process is shown in Figure 1 below.

2.1.3 Feature Selection

Since the features of URLs could be divided into three parts: Lexical-based, host-based and correlated-based, three groups of features are trained and tested to make comparison and assessment in this study. The first group will take only lexical-based features, such as Alphabets and lower or upper case letters in URL (Raja et al 2021). The second group will take all the words which are separated within the URL into account.

2.2 Machine Learning

2.2.1 Logistic Regression

Therefore, it is assumed that the data obey this LR distribution, and then use maximum likelihood

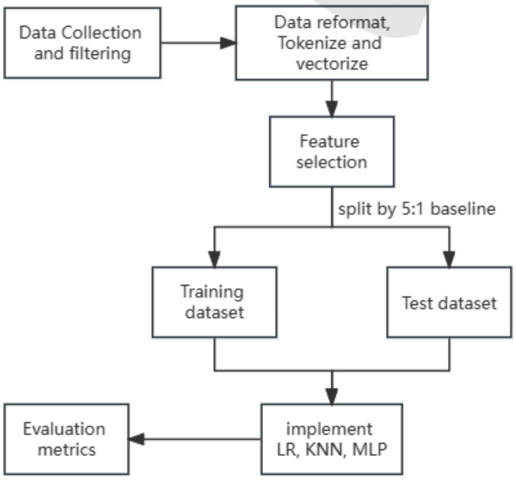

Figure 1: Process flow chart (Picture credit: Original).

estimation to estimate the parameters. Using this model to classify the malicious URLs into two classes, ‘malicious’ or ‘benign’. It comes with a modest computational burden, simplicity in comprehension and implementation. To implement this model, it is necessary to assume the training data at least could be fitted in logistic distribution which could be allowed to use maximum likelihood estimation to estimate the parameters. According to paper (Chiramdasu et al 2021 & Vanitha et al 2019), it has been proven that using logistic regression to classify URL into malicious and benign is available and sometimes better than other traditional machine learning algorithm such as K neighbors. One advantage of logistic regression model is its modest occupation and cost of computational resources which make this model relatively efficient in terms of processing and training time. Additionally, it is simple to implement logistic regression and it is easy to make comprehension, enabling users to easily understand and utilize the technique for predictive modeling tasks. In this research, it is highly expected that LR model will have overall good performances on extremely large dataset and hybrid feature combination as this model has such properties. Nevertheless, LR model probably not fit well based on selected feature groups.

2.2.2 Neural Network

An Artificial Neural Network (ANN), also referred to as Multi-layer Perceptron (MLP), usually consist with one input layer, one output layer and at least one hidden layer between the input and output layer. The more hidden layer it has, the more complex model it will be, taking more to train but it could also achieve higher accuracy of prediction. In A Aljofe et al’s study in 2020, the paper explored the availability and performance of using neural network on phishing URL detection and posted some features might be suitable for neural network. However, it does not take other types of malicious url into consideration and it has some noise of some sensitive words. In this paper, mainly use neural network model with 'Relu' and 'Logistic' activation to train and make prediction. Making neural network model as main comparison with LR and KNN to analyze and assess the performances and differences. Because the neural network should have better score on large dataset as this model possesses a strong capacity to extract information from big data and construct highly complex models. But this model will take longer time to train and require appropiate data pre-processing at the same time.

As the dataset contains over 20,000 data, choosing ‘adam’ (Jais et al 2019) will be the best optimizer as ‘adam’ has been proved as best optimizer for large training. However, with the limitation of computational resource, the more hidden layer with more neurons will make the cost of training much higher. Therefore, as testing several possible number of layers, it is better to apply two hidden layers with 5 neurons each to make MLP classifier have the best performance on the dataset. And for this research, the activation will test ‘Relu’ as default and ‘Logistic’ as comparison to choose the one with better performance.

2.2.3 K Neighbors Classifier

K-Nearest Neighbors (KNN) is a widely-used supervised classification algorithm known for its simplicity and easy to make adjustment on parameter. The principle behind KNN involves assigning a class label to an instance based on the categories of its nearest neighbors, determined by measuring their distances. Implementing KNN is straightforward, as it does not require parameter estimation or complex training procedures. It usually has good performance on classification. However, KNN is considered a lazy algorithm, as it involves extensive computations for classification. It requires scanning all training samples to calculate distances, leading to high memory usage and slow inference speed. The larger the dataset, the more time and resources will cost. In this research, applying KNN as a comparative model against other training models. The KNN model is expected to have better performance on classify malicious and benign URLs as this is only two classifications. However, this model might occupy large amount computation resources to train and predict due to the large volume of data. As mentioned in Shah’s study in 2020, LR usually outperforms KNN on large dataset and complex situation, but KNN could have better performance on selected feature group. Hence, it is meaningful to implement KNN model in this research as we choose lexical feature group to compare with hybrid feature group.

2.3 Evaluation Metrics

Four types of scores will be implemented as evaluation metrics: accuracy, F1, precision, recall. These four metrics will show the all-rounded scores of the model.

$$Precision = TP/(TP + FP) \tag{1}$$

$$Recall = TP/(TP+FN) \tag{2}$$

$$Accuracy=(TP+TN)/(TP+FP+TN+FN) \tag{3}$$

$$F1=2*Precision*Recall/(Precision+Recall) \quad (4)$$

There are four variables in the equation. The first one is True Positive (TP). In this research, the TP represents the total amount of malicious URLs classified match with the label of test data. The second one is True Negative (TN). The TP represents the total amount of good URLs which have been classified and matched with the label of test data. The third one is False Positive (FP). The FP represents the total amount of malicious URLs classified but the label of test data is good URLs. The last one is False Negative (FN). The FN represents the total amount of good URLs classified but the label of test data is malicious.

3 EXPERIMENTAL RESULTS AND DISCUSSION

The results involved two groups of features, the first feature group training is based on all correlated features, which vectorised all possible words and features separated by all kinds of punctuation marks. The final score is shown in Table 1.

Table 1: Scores of hybrid feature group.

Evaluation	LR	KNN	MLP
Accuracy	98.39	96.83	99.71
F1	98.18	96.49	99.67
Precision	98.61	95.98	99.69
Recall	97.80	97.80	99.54

The second feature group training is based on only lexical-based features, which mainly extracted length of URL and different part of URL separated by slash, the number of different letters and punctuation marks. The final result is shown in Table 2.

Table 2: Scores of lexical-based feature group.

Evaluation	LR	KNN	MLP
Accuracy	93.19	97.89	99.49
F1	94.57	98.11	99.56
Precision	93.79	97.80	99.38
Recall	93.21	98.76	99.44

The two groups of models achieved high scores after optimizing the parameter. However, it is easy for LR models to overfit and KNN underfit in the first group of correlated features. According to the learning curve of Figure 2, when the training data is under 50,000, the LR model shows high variance, and the model becomes fitting after 100,000 data to finally reach just right. For the KNN model, as testing all possible parameters of KNN from 5 neighbors to 50 neighbors, the model shows underfitting when the number of neighbors larger than 30.

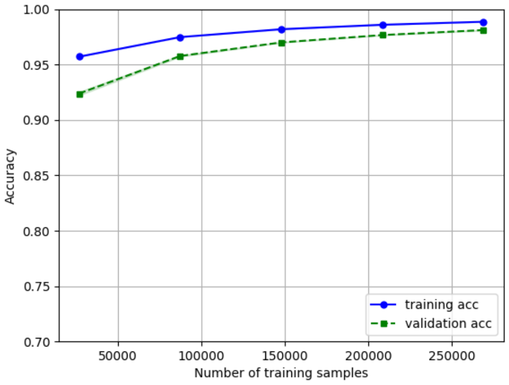

Figure 2: Learning curve of LR model (Photo/Picture credit: Original).

It is obvious to observe that MLP have the best performance in both two groups after optimizing the parameter and active function and the learning curve shows in Fig.3, which displays a good fitting and training of neural network model. Nevertheless, the score will have tiny change around 0.2% if reduce or increase the hidden layer and iteration epoch. This could probably be because the simple structure of URLs and limit combination of features which make no difference to allow deep learning algorithm to train the model.

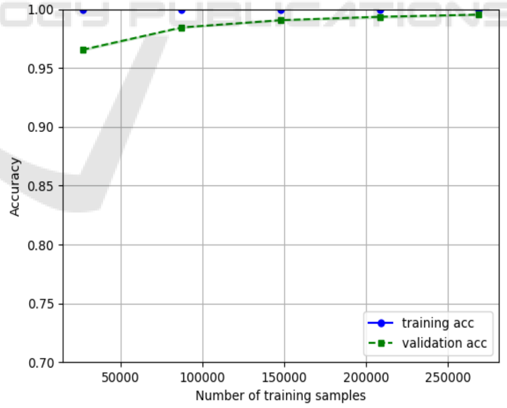

Figure 3: Learning Curve of MLP model (Photo/Picture credit: Original).

Calculating the average score of all four evaluation metrics in the two groups to make comparison, and the comparison result shows in Fig.4. The result shows MLP has the highest scores on both two groups of features. LR has better performance on correlated feature group, and KNN has better scores on only lexical feature group.

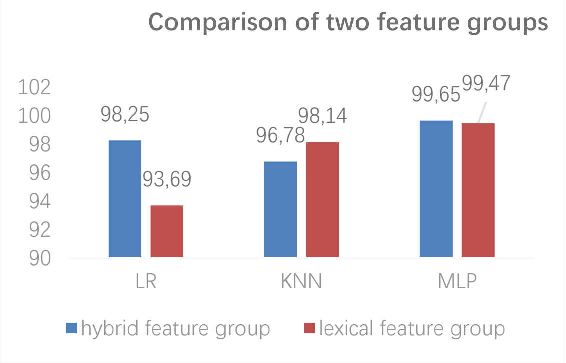

Figure 4: Comparison of two feature groups (Photo/Picture credit: Original).

However, the neural network has taken much more time to train and predict compared with the other two models, especially for the hybrid feature group. It is mainly because of the high degree of the feature after vectorizing, the degree of first feature group could reach over 30,000 after vectorising all the possible words and features of URLs, which make the efficiency of neural network extremely low.

4 CONCLUSION

In this study, three models underwent training using two distinct sets of features and a dataset comprising over 20,000 instances. The objective was to scrutinize their behaviors and ultimate performance, with the aim of determining which model is best suited for URL analysis and detection. The neural network model exhibited superior performance in evaluation scores compared to LR and KNN, but it was associated with the lowest efficiency. Conversely, KNN demonstrated strong overall performance in terms of both efficiency and accuracy when applied to the lexical-based feature group. The LR model is more suitable for complex feature groups and extremely large dataset as the algorithm takes less resources but high speed to train. However, this research only compares two types of features groups and needs more combination of features to compare deep learning algorithm with traditional machine learning model. And the dataset only comes from one resource which might lead to some bias of data. In the future, further studies are looking forward to seeking more groups of features, exploring more newly deep learning model and scratching more data from different resources to full-fill the analysis and evaluation.

REFERENCES

D. Sahoo D et al. Malicious URL detection using machine learning: A survey. arXiv preprint arXiv:1701.07179, (2017).

X. C. Do et al. Malicious URL detection based on machine learning. International Journal of Advanced Computer Science and Applications, 11(1) (2020).

B. B. et al. A novel approach for phishing URLs detection using lexical based machine learning in a real-time environment. Computer Communications, 175: 47-57 (2021).

G. Palaniappan et al. Malicious domain detection using machine learning on domain name features, host-based features and web-based features. Procedia Computer Science, 171: 654-661 (2020).

J. Kumar. Hybrid Feature-Based Machine Learning Method for Phishing URL Detection. 2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC). IEEE, 222-227 (2023).

URL 2016 | Datasets | Research | Canadian Institute for Cybersecurity | UNB https://www.unb.ca/cic/datasets/index.html (2016)

A. S. Raja et al. Lexical features based malicious URL detection using machine learning techniques. Materials Today: Proceedings, 47: 163-166 (2021).

R. Chiramdasu et al. Malicious url detection using logistic regression. 2021 IEEE International Conference on Omni-Layer Intelligent Systems (COINS). IEEE, 1-6 (2021).

N. Vanitha et al. Malicious-URL detection using logistic regression technique. International Journal of Engineering and Management Research, 9(6): 108-113 (2019).

A. Aljofe et al. An effective phishing detection model based on character level convolutional neural network from URL. Electronics, 9(9): 1514 (2020).

I. K. M Jais et al. Adam optimization algorithm for wide and deep neural network. Knowledge Engineering and Data Science 2.1 41-46 (2019).

Shah, Kanish, et al. "A comparative analysis of logistic regression, random forest and KNN models for the text classification." Augmented Human Research 5 (2020): 1-16.