Sakarya University Journal of Computer and Information Sciences
http://saucis.sakarya.edu.tr/

DOI: 10.35377/saucis…1273536 RESEARCH ARTICLE

Classification of Malicious URLs Using Naive Bayes and Genetic Algorithm

Murat Koca[1*], İsa Avcı[2], Mohammed Abdulkareem Shakir Al-Hayani[2]

[1]Van Yuzuncu Yil University, Faculty of Engineering, Department of Computer Engineering Van/Türkiye
[2]Karabuk University, Faculty of Engineering, Department of Computer Engineering, Karabuk/Türkiye

Corresponding author:
Murat Koca,
Van Yuzuncu Yil University,
Faculty of Engineering,
Department of Computer Engineering
E-mail address:
muratkoca@yyu.edu.tr

Submitted: 30 March 2023
Revision Requested: 17 April 2023
Last Revision Received: 22 May 2023
Accepted: 27 May 2023
Published Online: 24 June 2023

Citation: Koca M. et al. (2023). Classification of Malicious URLs Using Naïve Bayes and Genetic Algorithm. *Sakarya University Journal of Computer and Information Sciences*. 6 (2) https://doi.org/10.35377/saucis...1273536

ABSTRACT

The financial losses of vulnerable and insecure websites are increasing day by day. The proposed system in this research presents a strategy based on factor analysis of website categories and accurate identification of unknown information to classify safe and dangerous websites and protect users from the previous one. Probability calculations based on Naive Bayes and other powerful approaches are used throughout the website classification procedure to evaluate and train the website classification model. According to our study, the Naive Bayes approach was benign and showed successful results compared to other tests. This strategy is best optimized to solve the problem of distinguishing secure websites from unsafe ones. The vulnerability data categorization training model included in this datasheet had a better degree of precision. In this study, the best accuracy probability of 96% was achieved in Naive Bayes' NSL-KDD data set categorization.

Keywords: HTML, Malicious, Naive Bayes, Machine learning, URL, Neural Network

1. Introduction

Hackers use commercial websites and random advertisements to spread their malicious links [1]. Because heavy internet users are persuaded to assume that their participation will yield financial gains, they fall prey to scam schemes, such as those that sell fake traditional fake ads, promote counterfeit loans, or sell cheap goods. Adequate expertise in avoiding actual physical injury and websites that pose a potential danger to our safety is not required [2].

Advertising exists for a variety of reasons, but its ultimate purpose is to get people to click through to the related sites and advertisements so they may read the content. In 2019, Symantec published a report on Internet security in which the company explained the existence of extensive and successive attacks on companies to steal information and cause significant losses, as well as large threats to personal and bank accounts and the threat of victims through threatening messages to pay a certain ransom using a variety of methods. Symantec also indicated the presence of widespread threats to personal and financial accounts, as well as the danger of victims receiving ransom demands. Clicking on harmful links in deceptive and fraudulent advertisements directs people to hostile websites [3].

At the moment, the technique that is employed to attack a network is also becoming more severe, and the difficulty of safeguarding worldwide networks is developing at the same rapid pace as the economy. The market for network security is expected to begin exhibiting signs of expansion around the year 2021, as stated by the projections [4].

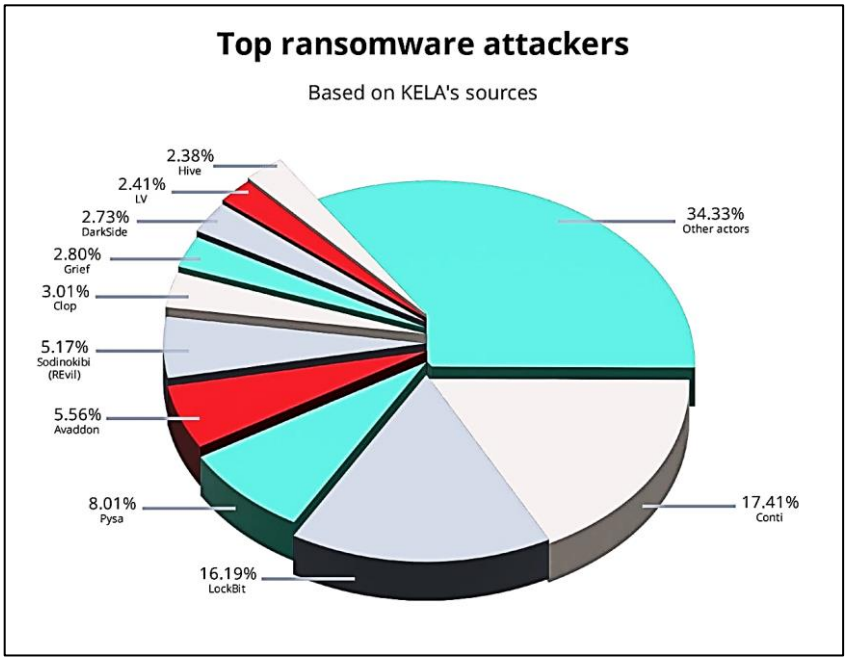

Figure 1. The proportion of ransomware attacks in malicious links [3].

The research shows that protecting and preserving the integrity of such networks has become an extremely important objective in light of the growing frequency with which cybercriminals attack networks. This is because cybercriminals are increasingly targeting networks. This is the case because con artists are placing a greater emphasis on networks, which has led to the current state of affairs in which the proposed system finds itself in. The majority of people will, throughout an ordinary day, navigate to a significant number of URLs on the internet. This activity is typically included as part of the activities that make up an average day. Unfortunately, within this ever-increasing multitude of URLs, there is now a considerable number of URLs that connect to dangerous websites. This is a very concerning trend. Because of the lightning-fast growth of the Internet, it is easy to mistakenly enter harmful URLs for those that are legitimate. Because it is possible to confound malicious URLs with genuine ones, making this error is not difficult to accomplish. As a result of this, it is much higher necessary to nurture the capabilities essential to discriminate between the two rapidly and exactly [5].

The goal of the academic community is to identify URLs that have the potential to be associated with fraudulent behavior and to do so, they make use of a wide variety of various benchmark models. The proposed system used a dataset that was comprised of URL occurrences so that the proposed system could assess the performance of K nearest neighbor KNN, support vector machine SVM, and Naive Bayes NB Tree. As a consequence of these studies, the proposed system realized that the application of this technology increased the accuracy with which the support vector machine and the KNN classified data. It was brought to our attention that decision Tree had the lowest degree of effectiveness when viewed in the context of the bigger picture. It has been claimed that a naïve Bayesian classifier may be used as a means of automatically classifying and determining which URLs have the potential to be fake. This could be accomplished through the use of a computer program. On a variety of benchmark data sets, the performance of the Navie Bayesian model, which was trained using probabilistic model learning, is superior to that of the support vector machine model [6]. This is the case even though the support vector machine model was trained using probabilistic model learning. This is the result of training having been done to improve the performance of the Naive Bayesian model. This is still the case even though the model of the support vector machine was constructed using probabilistic model learning. They developed a multi-stage filtering system that would detect potentially dangerous URLs by basing it on techniques that are associated with machine learning, which is a field of research that is abbreviated as ML [6].

It is based on techniques associated with deep learning, which is another acronym for deep learning. Because the classifier was trained with the critical threshold, it is now possible for the classifier to zero in on URLs at which it performs exceptionally well and pinpoints those. This is a direct effect of training the classifier with the critical threshold. Because of this, the classifier can function at its highest possible level of efficiency. If you discover that a certain group of classifiers is unable to accurately allocate a URL into one of their categories, you need to cast your vote with a number of the classifiers. In conclusion, it is vital to emphasize that the accuracy of spotting malicious URLs is enhanced when utilizing this method in comparison to the Bayesian model, the decision tree model, and the SVM model. This point cannot be emphasized enough. To correctly classify potentially hazardous URLs, logistic regression, neural networks, and three distinct iterations of the Naive Bayes approach were applied as analytical tools. According to the results of the study, the Naive Bayes strategies were the ones that had the highest success rate overall. Sheikh Shah Mohammad Motiur evaluated the efficacy of a large number of machine learning classifiers to determine whether or not they were capable of accurately identifying phishing URLs [7].

The metrics that he used to evaluate the effectiveness of these classifiers included the area under the receiver operating characteristic curve (AUC-ROC), accuracy, misclassification rate, and mean absolute error. When it comes to binary classification and feature sets that contain several different classes, stacking generalization provides more accurate results than random forest and multi-layer perceptron do. The impact that using a large number of machine learning models, in particular ML ensembles, to tackle the problem of locating phony URLs has. The random forest is the result of inheritance learning, and multiple metrics, including the recall rate, the accuracy rate, and the Area Under the Curve (AUC) value, have demonstrated that it is superior to the conventional ML model. In addition, the random forest is the product of inheritance learning. This state of affairs has arisen as a consequence of the fact that inheritance learning was the impetus behind the development of the random forest. Convolutional Neural Networks (CNN), Long Short-Term Memory Networks (LSTM), and Convolutional Neural Networks (CNN-LSTM) were the three distinct types of deep neural networks that were used in the process of identifying fake URLs [8].

However, to propose a YOLO-inspired multi-layer recursive convolutional neural network model for malicious URL identification, they did not conduct a comparison of the hidden layer and the number of neurons in the experiment. The abbreviations (CNN), (LSTM), and (CNN-LSTM) are the three that are most frequently utilized. The Text-RCNN and BRNN models, in addition to many other approaches, are not able to compete with the level of precision that can be accomplished by utilizing this method. During the study, each URL will have its length shortened in precisely the same manner until they are all the same length. This process will be repeated until there is no longer any distinction between any of them [9]. Working with longer URLs exposes you to a greater risk of losing data than doing the same activity with shorter URLs. There is a one-to-one relationship between the length of the URL and the degree of severity of this risk. The use of the benchmark machine learning model as a stepping stone in the process of developing more advanced feature engineering strategies to increase the rate at which potentially harmful URLs are detected is a possibility.

This would be done to boost the rate at which potentially harmful URLs are detected. A method was created for feature engineering that can change the spatial coordinates of the thing that is being generated in either a linear or nonlinear fashion, depending on what kind of change the proposed system wants to make. Both of these applications are viable options for making use of this technology. The recognition rates of KNN, linear support vector machine, and multi-layer perceptron are significantly improved when five separate spatial transformation models are utilized to generate and apply extra features to the classifier. This is because the classifier now contains a greater number of features. The generation of new capabilities and features is accomplished with the help of these five models.

The extraction of information from the text that is included within the URL is the primary focus of the great majority of the methods that may be employed in today's world to identify fake URLs. This holds for the vast majority of the different methods. The proposed system provides an association classification-based data mining method for identifying harmful URLs based on URLs and attributes acquired from online content. This method analyzes the relationships between the URLs and the attributes. This approach makes use of the URLs themselves, in addition to the attributes that may be extracted from the URLs. This strategy combines the usage of classification with that of association rules to accomplish what needs to be done. a set of instructions that, when combined, allow one to place things into categories and make relationships between those categories [10].

The proposed system first presents a weighted approach that extracts a fundamental set of characteristics for study, and then the proposed system evaluates machine learning algorithms based on how quickly and effectively they learn. This is done to further our understanding of the topic. This method collects lexical information from URLs and then analyzes it straightforwardly to locate links that can cause harm. The random forest algorithm and the KNN algorithm both have the potential to produce positive findings from the investigation. It was discovered that the pure lexical technique had the potential to enable rapid real-time determination of URLs in lightweight systems. This turned out to be the case [11].

This is accomplished by first collecting static lexical features from URL strings and then classifying them using an ensemble classification algorithm that has been trained by machine learning. This process is repeated until the desired result is achieved. This procedure is carried out numerous times until the desired outcome is accomplished. developed a method for detection that is wholly dependent on the lexical characteristics of the content they are looking to uncover. The convolutional neural network is now in a position to deliver a classification result that is more accurate because the URL strings have been gathered and processed. As a direct consequence of the increased precision of the classification, this is now conceivable. Developed a concatenated neural network technique model by combining Bi-Ind RNN and Caps Net to recognize malicious URLs, extract features (including vector and texture features at the character and word levels), and combine features [12].

Specifically, this model was created to recognize malicious URLs. Specifically, the identification of dangerous URLs was the motivation behind the creation of this approach. When a mixed neural network is used for the classification process, not only is there a significant rise in the speed with which potentially harmful URLs can be recognized but there is also a significant increase in the accuracy with which they can be identified. URL Net is a CNN-based deep neural network that was developed to determine whether or not a URL is fraudulent. The network was built to determine whether or not a URL is phony. The name of the network is derived from the term "universal resource locator," which describes its function. CNN and the word "CNN" are promoted, and the network is set up to manage the limits of manual feature engineering as well as the absence of visibility into the test URL [13].

2. Related Work

A lot of research has been done on the difficulty of categorizing a large number of websites currently available. The Naive Bayes classifier, which is commonly used to develop high-quality solutions to search problems using biologically inspired operators, can be combined with the Genetic Algorithm, which is commonly used to reduce the processing time by developing high-quality solutions to search problems using biologically inspired operators [14]. The Naive Bayes classifier would be able to develop high-quality solutions to search problems using biologically inspired operators. Using biologically inspired operators in this manner would allow the Naive Bayes classifier to generate high-quality solutions to search issues. In addition to that, the combination can cut down on the total amount of time that is needed for the process. They offer a variety of extra functions in addition to lexical and host-based capabilities. These features include the ability to enable or disable JS, the content of an HTML element, and a great deal more. Additional processing time is required for the categorization of web pages because there are a staggering 31 distinct attributes that can be utilized to accomplish this task [15].

On the other hand, it is possible to organize web pages. As a consequence of this, it is possible to classify websites according to a wide range of distinct characteristics. They used a range of tactics, such as recurrent feature switching and GA-based optimization, to achieve the desired level of accuracy, and they were successful in doing so [16]. The researchers observed that by employing a Naive Bayes classifier to identify websites based on URL properties, they were able to get an accuracy rate of 78% when utilizing this strategy. The results of the investigation led to this conclusion being drawn. This was a significant increase in accuracy in comparison to their earlier rates, which stood at 74% without GA and 87% with GA, respectively. This reflected a rise of 3% from the previous year. They are concentrating their efforts on discovering how people utilize websites, and the dataset that they are making use of is quite comprehensive. A key source of disappointment is the fact that the overall accuracy is lower than what was anticipated, even though the average number of recalls is higher than 88%. This is because the average number of recalls is higher than 88%.

This behavior involves pretending to be another website to steal sensitive information from its users. Even if the projects they're working on aren't specifically designed to find fraudulent websites, the strategies that they use to do so are significant in and of themselves. They were able to achieve ultimate accuracies of 89.1% and 88.5% with the Rotation Forest method, which proves that it is the best algorithm after undergoing a substantial amount of training and testing. After the installation of the feature ranking, they were successful in reaching an accuracy of 89% when utilizing MLP and an accuracy of 87.5 when utilizing REP Tree. Both of these results were achieved following the introduction of the feature ranking. It has been discovered that the utilization of multilayer perceptrons, which are one category of feedforward artificial neural networks, yields the most successful results that can be achieved in previous investigations. Because of this network, the number of dimensions that the data consists of will be cut down significantly. If the researchers use Rotation Forest on the entire training set, there is a good chance that they will achieve a significant improvement in their accuracy margin. The method of forest rotation would then be utilized in this scenario. This is because Rotation Forest is a method for supervised learning, which explains why these results were obtained. Rotation Forest beats other algorithms, such as Random Forest and Hyper Random Tree [2], in practical applications because it generates a bigger number of outcomes employing a lower number of trees. This is the primary reason why Rotation Forest is superior. This is the fundamental reason for the aforementioned advantage that it possesses in this regard.

To put it another way, the effectiveness of the Rotation Forest method is far higher. Nevertheless, the difficulty of their algorithms and the length of time that would be necessary to carry out their instructions will continue to be roadblocks. Support Vector Machine, J48, Naive Bayes, and Logistic regression are some of the approaches that should be utilized when searching websites for malware. Other methods that should be employed are Logistic regression and Naive Bayes. These are only a few of the available choices. They evaluated some different single-layer classifiers and concluded that J48 was the one that was the most effective at identifying websites that could contain malicious content. As a result of their additional research, they concluded that the XOR-aggregation cross-layer detection strategy is superior to the others because it rarely requires the utilization of the dynamic methodology [13]. This realization led them to the conclusion that the XOR-aggregation cross-layer detection strategy is superior to the others. After conducting the inquiry, they realized that this was the most likely explanation.

When applying the Naive Bayes classifier, it is necessary to bear in mind that the usage of multiple attributes does not result in particularly pleasing detection results. This is something that must be kept in mind at all times. It is imperative that this particular point not be ignored. Bear in mind that this is something that must be considered, as this must be done. Although it was possible to use a data aggregation cross-layer J48 classifier to achieve an astounding accuracy of 99.178%, the process took longer than four minutes, which may be uncomfortable for some customers. Because of the ever-increasing prevalence of social networks and the enormous amounts of data that they generate, researchers' curiosity has been piqued. This is because researchers have been able to see more data than ever before. In recent years, a significant amount of focus and research has been placed on a variety of topics, including the identification and filtering of spam, the localization of communities, and the dissemination of knowledge, to name just a few of these topics' specific manifestations.

Some different classification approaches, such as SMO, Naive Bayes, J48, and random forest, are utilized during the process of sorting spam emails into the right categories. These are but a few of the many different classification strategies that are put

into use. According to the results of the research, random forest performs much better than alternative classification approaches in terms of weighted precision (95.50%), recall (95.50%), accuracy (95.50%), and F-measure (95.50%). It was proposed that the S3D spam detection approach, which was cited in the previous sentence, could be employed instead of the semi-supervised spam detection method. S3D employs four unique lightweight classifiers to accomplish its goal of identifying spam tweets in real time. This is made possible by the platform's modular design. The authors of the article presented a strategy for recognizing Twitter spam that was dependent on the user's current mental state as the determining factor [4].

This strategy was utilized to determine the emotions that were mentioned in the Bengali text. It has been determined, based on the findings of the trials that were carried out, that the method that was proposed is capable of achieving an accuracy of 77.16 percent in the detection of two fundamental emotions (grief and happiness) in Bangla text. This conclusion was reached as a result of the findings that were obtained from the tests that were carried out. The results of the tests that were carried out served as the basis for this conclusion, which was arrived at as a result of those findings. In the course of these examinations, participants were tasked with locating instances of the aforementioned emotions within a Bangla text. According to the findings of an investigation that was carried out by Houshmand Craniometry using some different machine learning strategies on a dataset of SMS spam that was made available from the UCI Machine Learning repository[17], 10-fold cross-validation produced the highest level of accuracy[17]. This investigation was carried out using a dataset of SMS spam that was made available from the UCI Machine Learning repository [18]. It introduces a completely new supervised machine learning algorithm that is built on behavioral data as a tool for recognizing spam accounts in social networks. This algorithm is intended to be used as a way to eliminate unwanted accounts [8]. It is credited with developing the method, which may be found in [18]. Identifying accounts that are being used for spam requires the application of this method. They collected the data they required from Weibo and then used an ELM-based method to locate spam accounts among the user accounts they obtained.

The textual content, information regarding the user profile, and social interactions are the three types of attributes that should be chosen in the sequence presented here as part of this technique. Each of these three types of attributes is derived from a distinct source. This technique was implemented to identify user accounts that were being used to send spam. According to the source that was cited, the effectiveness of identifying spam SMS allegedly increased after the addition of a new content-based feature that was put into place. This was said by the source that was cited. The findings that were compiled from the application of a wide variety of classification strategies lend credence to the assertion that the suggested enhancements will lead to an increase in the level of precision achieved by SMS spam detection. These findings were compiled after applying the various classification strategies to a large number of messages. In addition, [10] discusses a spam detection system for social media that is both web-based and scalable.

The purpose of the system is to preserve the trustworthiness of social networks by warding off the creation of false posts and comments. Because of this, they were thus able to cluster vast volumes of data in a manner that was more efficient for them. The names of these algorithms, which are Decision Tree, KNN, Naive Bays, and Support Vector Machine, are how they are often referred to as (SVM). A training experiment simulation has been built as a means of emulating the progressive improvement that may be seen in individual spam filters. This improvement may be witnessed as the filters become more effective over time. To recreate the conditions of the training experiment, this has been carried out. It is possible that the training that was completed contributed to this progress. It was demonstrated that the SVM classification strategy was the most accurate of those that were used to evaluate whether or not the tone of a Bengali newspaper headline was negative or positive. The purpose of this evaluation was to determine whether or not the tone of the headline should be considered negative or positive. This analysis was conducted to determine whether or not the headline of a Bengali newspaper had a negative or a positive tone.

Logistic regression boosted tree, and support vector machine is just a few of the other methods of categorization that were applied here. The use of text processing software to perform semantic analysis and establish context comes highly recommended. They placed it to the test by utilizing a dataset that was open to the general public, did not include any information that was encrypted, and was genuine. They also integrated some additional well-established machine learning techniques, all of which have the potential to improve spam filtering in instant messaging as well as SMS. This was done in addition to the machine learning techniques already applied.

Malware is notoriously tough to eliminate due to its many manifestations and rapid dissemination. Listed below are our most important contributions to the field. The proposed system is currently constructing a model to identify URLs as either secure or dangerous using the Naive Bayes algorithm, a cutting-edge piece of technology that deviates from conventional methodologies.

3. Methodology

The only kind of file that can be loaded into the software at this time is a.csv. After doing individual tests on each of the features, the next step is to use Naive Bayes classifier to determine the five most important characteristics. In the meantime, these five characteristics would be used to evaluate which feature set had the greatest room for development. Following completion, the solution would be assessed using a test dataset. Figure 1 is a visual representation of the process.

3.1. Dataset

In the course of this particular experiment, the proposed system made use of the dataset that Kaggle made accessible to us for our research purposes as seen in Figure 2. The gathering of information was the most important task that needed to be accomplished. During our investigation on the internet, the proposed system came across some websites that contained links to a variety of other websites. Some of these other websites may contain content that is detrimental to users, and some of these other websites may also contain links to other potentially harmful websites [1].

The third step involved identifying URLs that were devoid of anything that would have contributed to the confusion that was being caused. The dataset was not only very simple to access but also did not require any sort of data processing or cleaning on our end because it was already in its finished form. In addition to that, the proposed system has produced a list of URLs, the vast majority of which go to malicious websites while others do not. Some of these URLs do not go to hazardous websites. Some of these URLs redirect to websites that are not malicious. Next, to determine which method is the most accurate in determining which URLs could potentially be harmful, a method of LR, Naive Bayes, and a CNN system was put through their paces. This study is to determine which method is the most accurate in making this determination.

This was done so that the proposed system could decide which method is the most trustworthy, therefore that is our motivation behind doing it. This was done to determine which method was the most accurate, and it was successful in that endeavor. This experiment takes a list of URLs as its input, and the degree to which it is successful or unsuccessful is determined by how accurately it guesses the locations of the web pages that are referred to by those URLs. This experiment accepts a list of URLs as its input. The experiment's input is a list of URLs if you care to provide one.

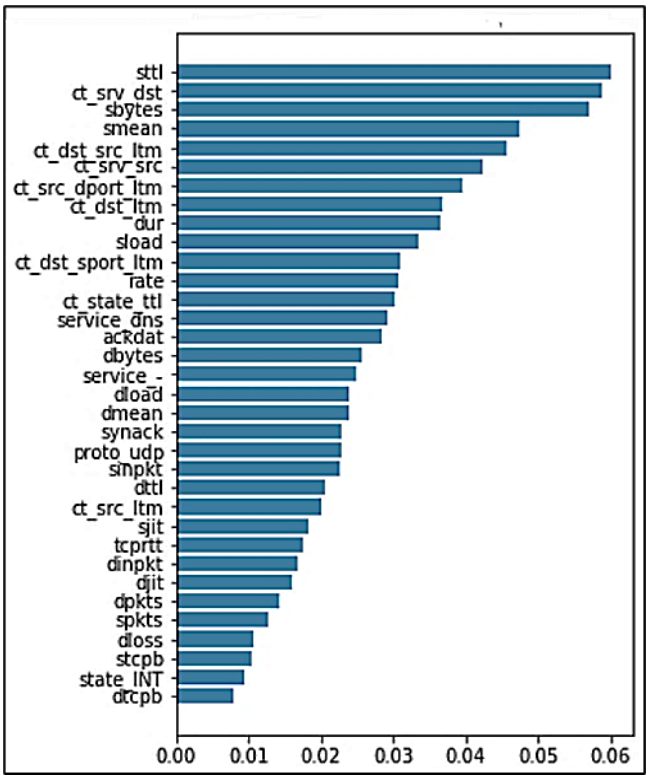

Figure 2. Features of the Dataset

3.2. Research Design

3.2.1. Naive Bayes Classifier

The Bayesian theorem and the premise of independence are the two foundations around which the Naive Bayes probabilistic model is constructed (the features being considered are unrelated to one another). Even though their seeming lack of complexity may be misleading, models that are trained with just two labels often perform rather well. Where x is a representation of the characteristic and C is the class being discussed. In light of this, the proposed system may describe C's conditional probability using the above as seen in Equation 1 when x is known.

$$\mathbf{P_r(C \mid x) = \frac{P_r(x|C)P_r(C)}{P_r(x)}} \quad (1)$$

3.2.2. Naive Bayes Classifier Probability-Based Prediction

Given that there are four distinct forms of harmful URLs, the software makes use of benign weight to filter them out. Naive Bayes assigns a probability of PPrr(CC00) to a benign URL and assigns PPrr(CC11), PPrr(CC22), PPrr(CC33), and PP(CC44) to each of four types of malicious URLs. The ultimate probability for good is PP(CC00), whereas for bad it is PP(CC11) as seen in Equations 2 and 3.

$$\mathbf{P(C_1) = P_r(C_1) + P_r(C_2) + P_r(C_3) + P_r(C_4)} \quad (2)$$

$$\mathbf{P(C_0) = wP_r(C_0)} \quad (3)$$

3.3. Proposed Methods

3.3.1. Choosing the Most Appropriate Features

A Naive Bayes classifier will be applied to each feature to identify a high-performing group. Then, the five characteristics with the greatest level of reliability would be selected. A Naive Bayes classifier would then randomly choose three of the aforementioned characteristics and utilize them to make a judgement. The conclusion is to choose the combination yielding the highest overall profit ratio.

3.3.2. Feature Extraction

During this research, the URL will be analyzed and interacted with in various ways. The data were first organized in a way that was easier to read. The first batch of feature data was produced by examining the content of the URL and making comparisons between possibly harmful and safe URLs. The evolutionary algorithm employs a method of feature extraction that helps in simplifying the feature vector, which in turn speeds up the processing of data. A "Genetic Algorithm (GA)" is a term that's used in the field of computer science to refer to the process of analyzing biological systems. The method of stochastic global search and optimization takes its inspiration from the natural setting in which the process of evolution takes place. An algorithm for a quick, exhaustive, and parallel global search serves as the foundation of this system.

This approach takes into account all of the potential solutions while avoiding the pitfalls associated with a locally optimal solution. In contrast, the genetic algorithm is not in any way restricted in any manner by considerations such as the requirement to keep function continuity or identify a given beginning point. Instead, the genetic algorithm is free to explore all possible starting points and preserve function continuity. Probabilistic optimization does not require the establishment of rules to automatically retrieve and navigate the optimal search space and to adapt the search direction as necessary. This is made possible by the fact that probabilistic optimization can adapt the search direction as needed.

The process began by seeding the population, encoding the attributes, and calculating the fitness of the individuals who would ultimately represent each chromosome. This allowed us to determine which individuals would best represent each chromosome. The crossover procedure was used to make children, and the chromosomes of the offspring were changed by the concept that the more fit an individual was, the better their chances were of being picked.

The crossover method was used to produce children after two members of the population were chosen at random from the population to take on the role of parents in the experiment. To develop a new population, one needs just to repeat the procedures from the approach that came before it. Last but not least, the proposed system might be modified to fit more closely the evaluation criteria established by the various frameworks (TPR, FPR, TNR, FNR). The following sequence presents the four rules:

The True Positive Rate is the percentage of potentially dangerous occurrences for which a positive detection was made by applying the analysis to the whole database of harmful examples as seen in Equation 4.

$$\mathbf{TP = \frac{N_{M\to M}}{N_{M\to M}+N_{M\to B}}} \quad (4)$$

4. Results and discussion

The dataset was evaluated using a total of three distinct approaches in its entirety. Following the completion of the data standardization, the proposed system proceeded to split into two distinct phases: the development phase and the assessment phase. When determining whether or not the results generated by a neural network are superior to those of the standard model, the performance of a conventional logistic regression model is used as a baseline. This allows one to determine whether or not the results generated by a neural network are superior. According to the findings as seen in Table 1, the accuracy score is 96%, the error rate is 0.04, and the recall score is 98%. It is successful in identifying safe websites with an F1 score of 95% as seen in Table 2. To determine whether or not a neural network can aid in the improvement of the issue, the proposed system employs one. The proposed system will begin by using the standard configuration of the SciKit learn class before moving on to exploring the various configuration choices in an attempt to get the maximum possible level of performance. The proposed algorithm results are shown in Table 1.

Table 1. Malicious website URL detection comparison using different methods.

Author	Algorithm	Accuracy	Error rate
Proposed algorithm	Modified NB	96%	0.04
Moruff Oyelakin et.al [5]	DT	88%	0.3
Subasi et.al [16]	KNN	87%	0.5
Jian et.al [18]	SVM	91%	0.2
Luo et.al [23]	CNN	93%	0.17

It can be seen with the classification of two classes where 0 means a successful classification of benign classes and 1 means a classification of malicious links as seen in Table 2.

Table 2. Malicious website URL identification using multinomial Naive Bayes: a confusion matrix.

Classes	Precision	Recall	F1-Score	Data
(0)	0.93	0.98	0.95	630
(1)	0.72	0.43	0.54	83
Accuracy	0.90	0.90	0.91	713
Macro avg.	0.82	0.71	0.75	713
Weighted	0.90	0.91	0.90	713

ML model performance over binary classification is shown comparatively on Naive Bayes, Decision Tree, KNN, Logistic Regression, and Random Forest algorithms as seen in Figure 3. In this experiment, Naive Bayes came to the fore as the most successful machine learning model.

5. Conclusion

The identification of potentially harmful URLs is one of the most important processes involved in the process of ensuring the safety of cybersecurity software. There is reason to be optimistic about the potential of machine learning algorithms The study was conducted by investigating the use of AI algorithms in the process of determining whether or not URLs might contain malicious content. The purpose of this study was to investigate the application of AI algorithms to the process of identifying whether or not URLs may include dangerous content. The results show a 98% recall percentage, an accuracy rate of 96%, and an error rate of 0.04. In this study, the proposed system was able to categorize potentially damaging URLs using

Logistic Regression, Neural Networks, and multiple Naive Bayes Algorithms. This allowed us to determine which URLs posed the greatest risk to users. Because the proposed model can identify which URLs constituted the biggest threat. When applied to the difficult distribution dataset, the results show that the Naive Bayes strategy performed noticeably better than both the logistic regression and neural network approaches. The proposed system has it in our plans to, among other things, do research on new datasets and experiment with a wide range of different machine-learning approaches.

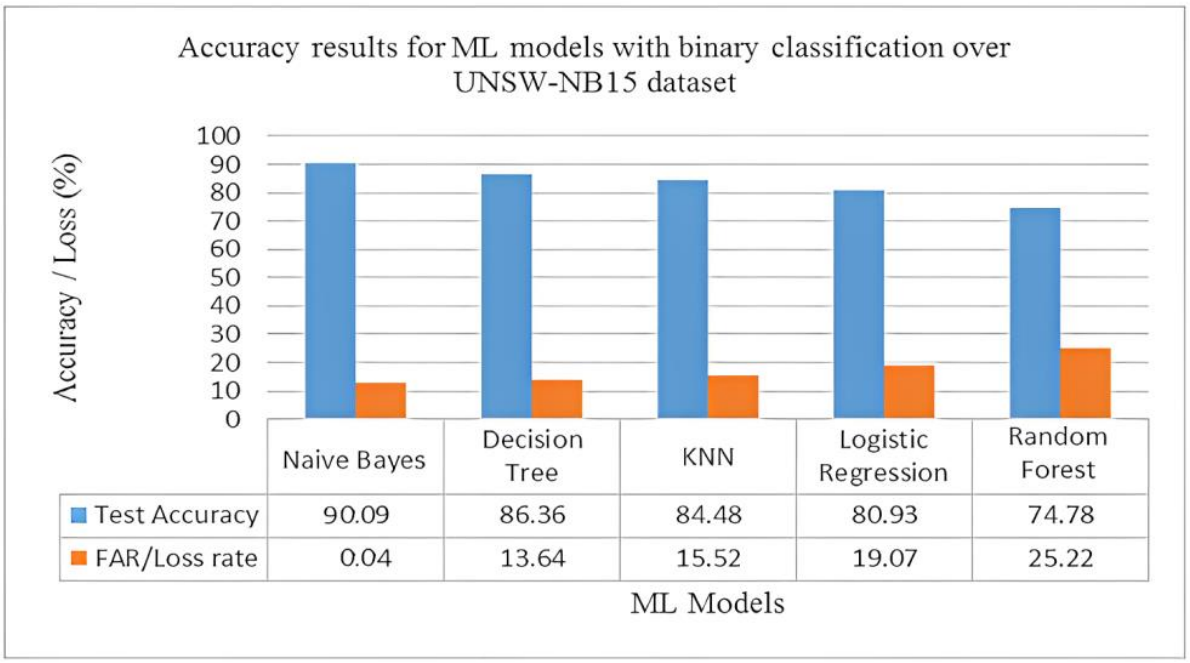

Figure 3. ML model performance over binary classification

References

[1] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set," *Submitted to Second IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, 2009.

[2] A. Sharma and A. Thakral, "Malicious URL classification using machine learning algorithms and comparative analysis," Advances in Intelligent Systems and Computing, vol. 1090, pp. 791–799, 2020, doi: 10.1007/978-981-15-1480-7_73/COVER.

[3] K. U. Santoshi, S. S. Bhavya, Y. B. Sri, and B. Venkateswarlu, "Twitter Spam Detection Using Naïve Bayes Classifier," Proceedings of the 6th International Conference on Inventive Computation Technologies, ICICT 2021, pp. 773–777, Jan. 2021, doi: 10.1109/ICICT50816.2021.9358579.

[4] T. Islam, S. Latif, and N. Ahmed, "Using Social Networks to Detect Malicious Bangla Text Content," 1st International Conference on Advances in Science, Engineering and Robotics Technology 2019, ICASERT 2019, May 2019, doi: 10.1109/ICASERT.2019.8934841.

[5] A. Moruff Oyelakin, O. Akinyemi Moruff, A. Olasunkanmi Maruf, and A. Tosho, "Performance Analysis of Selected Machine Learning Algorithms for the Classification of Phishing URLs Machine Learning Techniques in building Predictive Models for COVID-19 View project Investigation of MANET security protocols and optimisation View project Performance Analysis of Selected Machine Learning Algorithms for the Classification of Phishing URLs", Accessed: Jan. 05, 2023. [Online]. Available: https://www.researchgate.net/publication/345161822

[6] Maciej Serda et al., "Synteza i aktywność biologiczna nowych analogów tiosemikarbazonowych chelatorów żelaza," Uniwersytet śląski, vol. 7, no. 1, pp. 343–354, 2013, doi: 10.2/JQUERY.MIN.JS.

[7] T. Wu, Y. Xi, M. Wang, and Z. Zhao, "Classification of Malicious URLs by CNN Model Based on Genetic Algorithm," Applied Sciences 2022, Vol. 12, Page 12030, vol. 12, no. 23, p. 12030, Nov. 2022, doi: 10.3390/APP122312030.

[8] R. Rajalakshmi, S. Ramraj, and R. Ramesh Kannan, "Transfer learning approach for identification of malicious domain names," Communications in Computer and Information Science, vol. 969, pp. 656–666, 2019, doi: 10.1007/978-981-13-5826-5_51/COVER.

[9] G. Wejinya and S. Bhatia, "Machine Learning for Malicious URL Detection," Advances in Intelligent Systems and Computing, vol. 1270, pp. 463–472, 2021, doi: 10.1007/978-981-15-8289-9_45/COVER.

[10] F. Alzubaidi, "DETECT MALWARE URL USING NAIVE BAYES ALGORITHM."
[11] A. E. El-Din, E. El-Din Hemdan, and A. El-Sayed, "Malweb: An efficient malicious websites detection system using machine learning algorithms," ICEEM 2021 - 2nd IEEE International Conference on Electronic Engineering, Jul. 2021, doi: 10.1109/ICEEM52022.2021.9480648.
[12] S. Wang, Y. Wang, and M. Tang, "Auto Malicious Websites Classification Based on Naive Bayes Classifier," Proceedings of 2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education, ICISCAE 2020, pp. 443–447, Sep. 2020, doi: 10.1109/ICISCAE51034.2020.9236912.
[13] S. Wang, Y. Wang, and M. Tang, "Auto Malicious Websites Classification Based on Naive Bayes Classifier," Proceedings of 2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education, ICISCAE 2020, pp. 443–447, Sep. 2020, doi: 10.1109/ICISCAE51034.2020.9236912.
[14] W. Fadheel, W. Al-Mawee, and S. Carr, "On Phishing: URL Lexical and Network Traffic Features Analysis and Knowledge Extraction using Machine Learning Algorithms (A Comparison Study)," 2022 5th International Conference on Data Science and Information Technology, DSIT 2022 - Proceedings, 2022, doi: 10.1109/DSIT55514.2022.9943832.
[15] C. Liu and G. Wang, "Analysis and detection of spam accounts in social networks," 2016 2nd IEEE International Conference on Computer and Communications, ICCC 2016 - Proceedings, pp. 2526–2530, May 2017, doi: 10.1109/COMPCOMM.2016.7925154.
[16] Subasi, A.; Balfaqih, M.; Balfagih, Z.; Alfawwaz, K. A comparative evaluation of ensemble classifiers for malicious webpage detection. Procedia Comput. Sci. 2021, 194, 272–279.
[17] Sayamber, A.B.; Dixit, A.M. Malicious URL detection and identification. Int. J. Comput. Appl. 2014, 99, 17–23.
[18] Jian, L.; Gang, Z.; Yunpeng, Z. Design and implementation of malicious URL multi-layer filtering detection model. Inf. Netw. Secur. 2016, 1, 6.
[19] Vundavalli, V.; Barsha, F.; Masum, M.; Shahriar, H.; Haddad, H. Malicious URL detection using supervised machine learning techniques. In Proceedings of the 13th International Conference on Security of Information and Networks, Merkez, Turkey, 4–7 November 2020; pp. 1–6.
[20] Rahman SS, M.M.; Islam, T.; Jabiullah, M.I. PhishStack: Evaluation of stacked generalization in phishing URLs detection. Procedia Comput. Sci. 2020, 167, 2410–2418.
[21] Zeyu, L.; Yong, S.; Zhi, X. Malicious URL recognition based on machine learning. Commun. Technol. 2020, 53.
[22] Pham TT, T.; Hoang, V.N.; Ha, T.N. Exploring efficiency of character-level convolution neuron network and long short-term memory on malicious URL detection. In Proceedings of the 2018 VII International Conference on Network, Communication and Computing, Taipei City, Taiwan, 14–16 December 2018; pp. 82–86.
[23] Chen, Z.; Liu, Y.; Chen, C.; Lu, M.; Zhang, X. Malicious URL detection based on improved multilayer recurrent convolutional neural network model. Secur. Commun. Netw. 2021, 2021, 9994127.
[24] Li, T.; Kou, G.; Peng, Y. Improving malicious URLs detection via feature engineering: Linear and nonlinear space transformation methods. Inf. Syst. 2020, 91, 101494.
[25] Kumi, S.; Lim, C.H.; Lee, S.G. Malicious URL detection based on associative classification. Entropy 2021, 23, 182.
[26] Raja, A.S.; Vinodini, R.; Kavitha, A. Lexical features based malicious URL detection using machine learning techniques. Mater. Today: Proc. 2021, 47 Pt 1, 163–166.
[27] Joshi, A.; Lloyd, L.; Westin, P.; Seethapathy, S. Using lexical features for malicious URL detection—A machine learning approach. arXiv 2019, arXiv:1910.06277.
[28] Kang, C.; Huazheng, F.; Yong, X. Malicious URL identification based on deep learning. Comput. Syst. Appl. 2018, 27, 27–33.
[29] Yuan, J.T.; Liu, Y.P.; Yu, L. A novel approach for malicious URL detection based on the joint model. Secur. Commun. Netw. 2021, 2021, 4917016.
[30] Le, H.; Pham, Q.; Sahoo, D.; Hoi, S.C. URLNet: Learning a URL representation with deep learning for malicious URL detection. arXiv 2018, arXiv:1802.03162.
[31] Yuan, J.; Chen, G.; Tian, S.; Pei, X. Malicious URL detection based on a parallel neural joint model. IEEE Access 2021, 9, 9464–9472.
[32] Zhao, G.; Wang, P.; Wang, X.; Jin, W.; Wu, X. Two-dimensional code malicious URL detection method based on decision tree. Inf. Secur. Technol. 2014, 5, 36–39.
[33] Liu, C.; Wang, L.; Lang, B.; Zhou, Y. Finding effective classifier for malicious URL detection. In Proceedings of the 2018 2nd International Conference on Management Engineering, Software Engineering and Service Sciences, Wuhan, China, 13–15 January 2018; pp. 240–244.
[34] Lin, H.L.; Li, Y.; Wang, W.P.; Yue, Y.L.; Lin, Z. Efficient malicious URL detection method based on segment pattern. Commun. J. 2015, 36, 141–148.
[35] Subasi, A.; Balfaqih, M.; Balfagih, Z.; Alfawwaz, K. A comparative evaluation of ensemble classifiers for malicious webpage detection. Procedia Comput. Sci. 2021, 194, 272–279.
[36] Sayamber, A.B.; Dixit, A.M. Malicious URL detection and identification. Int. J. Comput. Appl. 2014, 99, 17–23.
[37] Jian, L.; Gang, Z.; Yunpeng, Z. Design and implementation of malicious URL multi-layer filtering detection model. Inf. Netw. Secur. 2016, 1, 6.

[38] Vundavalli, V.; Barsha, F.; Masum, M.; Shahriar, H.; Haddad, H. Malicious URL detection using supervised machine learning techniques. In Proceedings of the 13th International Conference on Security of Information and Networks, Merkez, Turkey, 4–7 November 2020; pp. 1–6.
[39] Rahman SS, M.M.; Islam, T.; Jabiullah, M.I. PhishStack: Evaluation of stacked generalization in phishing URLs detection. Procedia Comput. Sci. 2020, 167, 2410–2418.
[40] Zeyu, L.; Yong, S.; Zhi, X. Malicious URL recognition based on machine learning. Commun. Technol. 2020.
[41] Pham TT, T.; Hoang, V.N.; Ha, T.N. Exploring efficiency of character-level convolution neuron network and long short-term memory on malicious URL detection. In Proceedings of the 2018 VII International Conference on Network, Communication and Computing, Taipei City, Taiwan, 14–16 December 2018; pp. 82–86.
[42] Chen, Z.; Liu, Y.; Chen, C.; Lu, M.; Zhang, X. Malicious URL detection based on improved multilayer recurrent convolutional neural network model. Secur. Commun. Netw. 2021, 2021, 9994127.
[43] Li, T.; Kou, G.; Peng, Y. Improving malicious URLs detection via feature engineering: Linear and nonlinear space transformation methods. Inf. Syst. 2020, 91, 101494.

Conflict of Interest Notice
The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical Approval and Informed Consent
It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Availability of data and material
Not applicable.

Plagiarism Statement
This article has been scanned by iThenticate ™.