

International Journal of
Engineering Research and Science & Technology



ISSN : 2319-5991

www.ijerst.com

Email: editor@ijerst.com or editor.ijerst@gmail.com

ENHANCED MALICIOUS WEBSITE DETECTION THROUGH MULTI-MODAL FEATURES AND CONVOLUTIONAL NEURAL NETWORKS

¹ Satika Praveen, MCA Student, Department of MCA

² K Muddu Swamy, M.Tech, Assistant Professor, Department of MCA

¹² Dr KV Subba Reddy Institute of Technology, Dupadu, Kurnool

ABSTRACT

Web apps are now widely used in many different business domains and are vital tools for billions of people in their everyday lives. Unfortunately, a lot of these programs are harmful, which poses a serious risk to Internet users as they may spread spam, install malware, and steal private data. Because of the intricacy of extracting representative features, the vast amount of data, the dynamic nature of dangerous patterns, the stealthiness of assaults, and the limits of conventional classifiers, detecting malicious websites by online content analysis is useless. Static Uniform Resource Locators (URL) characteristics may often provide you instant information about a website without requiring you to load its content.

However, complicated feature extraction, large data volumes, changing attack patterns, and the limits of conventional classifiers sometimes make it difficult for current systems to identify malicious web apps using online content analysis. It is inadequate to rely just on lexical URL properties, which might result in incorrect classifications. In order to improve the efficacy of dangerous website identification, this research suggests a multimodal representation strategy that combines textual and image-based information. While picture features are useful for identifying more generic harmful patterns, textual features help the deep learning model comprehend and convey specific semantic information pertaining to attack patterns. By doing this, it may be possible to identify patterns in picture format that are obscured in text format. To extract the hidden features from both textual and image-represented information, two Convolutional Neural Network (CNN) models were built. Both models' output layers were merged and fed into a classifier that

uses artificial neural networks to make decisions. The usefulness of the suggested model in comparison to other models is shown by the results. While the false positive rate decreased by 1.5%, the overall performance as measured by the Matthews Correlation Coefficient (MCC) increased by 4.3%.

I. INTRODUCTION

Over 1.11 billion websites exist worldwide, and their number has been increasing rapidly in recent years, according to the Siteefy website [1]. There are 252,000 new websites made per day (REF Please). It is anticipated that there will be over 50 billion online pages as of May 9, 2023. Despite the fact that the majority of websites are made with good intentions, many of them are harmful [2]. Malicious websites are made with the intention of hurting users in some manner, either by infecting their computers with malware or stealing their personal data. They may be used for denial of service assaults, spam distribution, phishing, and malware distribution [3]. An estimated 12.8 million harmful websites exist on the internet, according to Google's extensive investigation [4]. Additionally, 18.5 million websites contain dangerous malware, according to writers in [5]. Due to the creation of new dangerous websites and the removal of existing ones, this figure is always fluctuating.

several studies have been conducted on the identification of malicious websites, and several solutions have been proposed. The following are included: [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23]. For many organisations, the most popular approach is the blacklist [24]. It updates slowly, however, since bad actors may simply change the URLs of their existing

websites or create new ones to get around blacklists. Because of this, blacklist-based systems find it challenging to stay up to date with the constantly evolving harmful website landscape [25], [26].

Many researchers have used machine learning approaches to identify dangerous websites in order to overcome the constraints of blacklisting. [27], [28], [29], scripts [15], [16], HTTP/s response [29], [30], URLs [6], [7], [8], [9], [10], [11], [12], [13], [14], [31], [32], [33], domain names [25], [34], [35], network traffic statistics [34], [36], and digital certificates [26] are all examples of the characteristics that these methods extract. To categorise websites as harmful or benign, a variety of machine learning methods were used, including logistic regression, random forests, decision trees, and support vector machines [28], [32]. The selection of features determines how successful machine learning techniques are [13], [14], [17], [18], [19], [20], [21], [22], and [23]. However, since malicious code is always evolving, attackers use obfuscation methods, the amount of data that must be analysed is enormous, and attacks are becoming more complicated, it is difficult to extract useful information. Regretfully, large and complicated datasets cannot be effectively classified using typical machine learning techniques. To enhance detection performance, however, efficient feature engineering is needed.

Large and complicated datasets may effectively have representative characteristics extracted by deep learning algorithms. Since it can automatically learn features from website text data, they can automatically extract useful features without the need for any human feature engineering. Commonly described techniques for detecting dangerous software included Convolutional Neural Networks (CNN) [22], Recurrent Neural Networks (RNN) [23], and attention processes. A lot of deep learning models are built using

characteristics that are taken from the content of websites. However, the dynamic nature of web material, the deployment of anti-scraping technologies to identify and stop automated scrapers, and the always changing nature of online dangers make it difficult to get big and varied datasets from website content for training deep learning models. To access material on some websites, user sessions and authentication are necessary. Simulating user behaviours, such as signing in, may be a part of scraping such websites. The structure and style of websites are constantly changing, thus scraping programs must be updated and maintained often to guarantee proper operation. Furthermore, for devices with limited resources, such Internet of Things devices, extracting representative characteristics from the online content may not be efficient. While content-based features are useful for identifying a wide range of threats, they are ineffective and inefficient for identifying sophisticated malicious websites.

It seems that the URL-based capabilities are a decent substitute for the online content features. Numerous studies evaluate the performance of models built with both features, and URL-based features consistently outperform the others. The majority of current research, however, only uses the lexical characteristics that are taken from URLs. Sparse feature vectors are created because lexical characteristics contain little semantic information. To enhance detection performance, several studies integrate digital certificates with URL characteristics. Since malicious websites often employ self-signed certificates or lack legitimate certificates, certificate analysis is a helpful way to determine a website's credibility. Digital certificate analysis may show if a website is using encryption, which is a standard practice for trustworthy websites. Not all websites, however, utilise digital certificates; some could use certificates that are self-signed or from less respectable Certificate Authorities (CAs). It may be difficult to extract

meaningful and pertinent features from certificates for machine learning models, and choosing the appropriate features is essential for successful detection. Digital certificates may also be incorrectly set, expire, or change often, which might result in a significant number of false warnings. In conclusion, complicated feature extraction, large data volumes, changing attack patterns, and the limitations of conventional classifiers make it difficult for current systems to identify malicious web apps using online content analysis. It is inadequate to rely just on lexical URL properties, which might result in incorrect classifications.

This work suggests a unique multimodal representation technique that combines textual and image-based elements to improve the identification of fraudulent websites in order to overcome these difficulties. This method makes use of the advantages of both modalities: picture features identify more general hostile visual signals, while textual features record specific semantic information pertaining to assault patterns. Through picture analysis, hidden patterns within written information may become apparent.

Two Convolutional Neural Networks (CNNs) are used in the suggested method: one for textual data and another for visual features. For better decision-making, their outputs are then pooled and fed into a classifier that uses artificial neural networks. Our findings show that the suggested model is better than the current methods. Our multimodal technique effectively detects malicious online apps, as shown by a 1.5% decrease in the false-positive rate and a 4.3% rise in the Matthews Correlation Coefficient (MCC).

The following were the contributions made by this study:

1. Malicious website identification becomes more thorough when DNS-derived characteristics are combined with URL-based ones. By

providing useful contextual knowledge on domain behaviour and infrastructure, this synergy strengthens the assessment of website security and authenticity, leading to a more thorough and sophisticated method of spotting fraudulent websites.

2. To describe a complete feature set, the research presents a multimodal representation technique that makes use of both textual and image-based characteristics. While picture features are useful for identifying more generic harmful patterns, textual features help the deep learning model comprehend and convey specific semantic information pertaining to attack patterns.

3. To uncover hidden characteristics in the textual and visual representations, create two Convolutional Neural Network (CNN) models.

4. To discover the connections between the hidden characteristics that the CNN models were able to extract, a second deep learning classifier was built. By using deep learning methods to integrate and use both textual and visual information for more efficient malicious website identification, this method improves the field.

The structure of the paper is as follows. The relevant literature is reviewed in Section II, and the suggested remedy is thoroughly explained in Section III. The experimental design is covered in Section IV, and the findings and comments are shown in Section V. The study is concluded in Section VI, which also addresses its limits and future directions.

II. LITERATURE SURVEY

A review of XSS vulnerability exploitation and detection techniques,"

W. Chen, X. Zhang, B. Zhang, and M. Liu

Online security is becoming more and more crucial as online applications proliferate. One kind of popular injection web vulnerability is the cross-site scripting vulnerability, or XSS for short. Web applications' business data may be altered, read, and deleted, user sessions can be taken over, malicious software can be inserted

into web apps, and victims can be controlled to attack other targeted servers. The taxonomy of XSS is covered in this article, along with the building of a sample website that illustrates the assault procedures of typical XSS exploitation situations. Along with comparing and analysing previous XSS detection research findings, the report categorises them into three groups based on various methods. Static analysis techniques, dynamic analysis methods, and hybrid analysis methods are the three types. The study categorises 30 detection techniques into the three aforementioned groups, compares them overall, outlines their advantages and disadvantages, and identifies the different kinds of XSS vulnerabilities. The study concludes by discussing potential strategies to stop XSS vulnerabilities from being used.

"A survey of emerging cybersecurity threats,"

S. Nepal and J. Jang-Jaccard,

The exponential expansion of Internet connectivity has resulted in a notable rise in cyberattacks, often with severe and catastrophic outcomes. Malware is the most common weapon used to carry out harmful intentions in cyberspace, either by taking use of special features of new technology or by exploiting flaws that already exist. The cybersecurity community has seen the creation of more creative and potent malware defence measures as an essential need. In order to help do this, we first provide a summary of the most often exploited flaws in the current network, software, and hardware layers. Critiques of current state-of-the-art mitigation strategies are then presented, along with an explanation of their effectiveness or shortcomings. Next, we go into new attack trends in cutting-edge fields including critical infrastructure, cloud computing, social media, and smartphone technology. We conclude by outlining our conjectural findings about potential avenues for further study. The crown All rights reserved. Copyright (c) 2014 Elsevier Inc.

An examination of web-based malware, or "The Ghost in the Browser,"

N. Provos, N. Modadugu, D. McNamee, P. Mavrommatis, and K. Wang,

An underground economy that infects hosts with malware or adware for financial benefit has emerged, targeting computer users as more people connect to the Internet and carry out their everyday tasks online. Regretfully, an attacker may identify weaknesses in the user's apps and compel the download of several malware files with only one visit to an infected website. Often, this virus gives the attacker complete access over the infected devices, which enables them to exfiltrate private data or install programs that enable remote host control. We think that this kind of behaviour is comparable to how we traditionally think about botnets. The primary distinction, however, is that web-based malware attacks are pull-based, which results in a weaker command feedback loop. We identify the four most common methods of injecting malicious information into well-known websites—web server security, user-generated content, advertising, and third-party widgets—in order to describe the nature of this emerging threat. We provide instances of online abuse for each of these domains. Presenting the current status of malware on the Web and highlighting the significance of this growing menace are our goals.

"Malicious URL detection using machine learning techniques based on lexical features,"

R. Vinodini, A. Kavitha, and A. S. Raja,

The most advanced cyberattack method used by cybercriminals is the creation and dissemination of malicious domain names or URLs via popups, emails, and other communications. Websites designed to distribute malware, viruses, worms, etc. once a person visits them are known as malicious URLs. The attack's primary goal is to either install malware on the victim's computer or steal user passwords and victim information. Therefore, the system that should identify malicious URLs and stop the assault has to be modified. Although several approaches are proposed by researchers, machine learning-based

detection techniques outperform them. The lightweight approach, which just incorporates the URL's lexical properties, is presented in this study. The outcome indicates that, in terms of accuracy, the Random Forest classifier outperforms the other classifiers.

"A comparative analysis of ensemble classifiers for detecting malicious webpages,"

C. Alfawwaz, Z. Balfagih, M. Balfaqih, and A. Subasi

One of the primary causes of criminal activity on the Internet is the creation or manipulation of malicious webpages for use as attack tools. Therefore, it is crucial to identify these websites and stop end users from visiting them. Searching through a blacklist, which is a list of URLs deemed dangerous from the viewpoint of users, is the foundation of traditional malicious webpage detection systems. However, because of their computational and technological constraints, these methods have substantial false-negative rates, particularly when used in conjunction with the aforementioned advanced assaults. Therefore, by methodically examining a collection of attributes that mirror the traits of a malicious website, machine learning approaches have been used to categorise sites. The prediction accuracy of several ensemble methods and machine learning classification algorithms is compared in this research. The comparative research uses a data set of 5000 URL occurrences with 189 distinct attributes. The findings indicate that Support Vector Machine (SVM) is the most accurate classification method in MultiBoost and Adaboost, but K-Nearest Neighbour (k-NN) is the most accurate method in bagging and random subspace.

III. SYSTEM ANALYSIS AND DESIGN EXISTING SYSTEM

Researchers have proposed three primary methods for classifying malicious URLs: URL-based, content-based, and blacklist [11], [32]. Numerous methods were put forward to build the detection classifiers, including machine learning approaches and heuristic principles derived from

professional expertise. Effective malicious URL detection is still a work in progress, though. The collected characteristics and the machine learning methods used to build the detection classifier have an impact on how well the most current malicious website detection solutions function. The authors of [32] provided a thorough literature analysis that addresses a range of machine learning-based methods for identifying fraudulent URLs, taking into account factors including datasets, feature types, detection technologies, and constraints. Research trends for malicious website detection systems include the combination of deep learning approaches with extracted characteristics. A blacklist of harmful URLs, like the Google Safe Web Browsing Tool, was created using the professional experience heuristic criteria [37]. However, since threats are always changing and regular database updates are required to identify the developing danger, blacklist solutions are useless for detecting malicious URLs.

In order to identify harmful information on websites, several researchers have used feature extraction approaches. For representation, natural language processing has been widely used. However, since attackers' methods are always changing, malicious website content is complicated, and these patterns become dynamic and covert, which reduces the accuracy of detection. For instance, the authors of [38] looked at how rogue websites use different web spam strategies to avoid detection. The goal is to provide a practical way to identify and stop rogue websites that use tactics like content hiding, hidden Iframes, and redirection spam. As a result, the research focusses on taking screenshots of websites from the viewpoint of the user and classifying them using a convolutional neural network. Nevertheless, the solution's ability to identify spam tactics is limited. Furthermore, since websites are dynamic, the functionality that relies on screenshots of the loaded page may be risky and incomplete.

In order to classify the HTTP/s answers, the authors in [27] gathered features and used a variety of feature transformation and selection strategies. These traits are dynamic, however, and may be obscured by encoding and encryption techniques, which can make the detection classifier useless. Many studies concentrated on deep learning methods, even if machine learning algorithms were often utilised to build the detection classifier. Effective categorisation is achieved by deep learning's ability to precisely identify comparable patterns discovered during training. However, it might be difficult to extract useful characteristics for categorisation since online material is very dynamic and may be encoded or encrypted to conceal dangerous tendencies.

Less dynamic URL characteristics hold promise for precise malicious domain identification. This is due to the fact that benign domains are made by people, but harmful domains are produced by algorithms. As a result, harmful URLs can have more noticeable properties than those taken from the content, which might be obscured or encrypted to trick the learning process. The authors of [38] concentrated on identifying harmful URLs that are produced by algorithms. They speculate that the dangerous URLs are automatically generated by attackers or malicious bots. As a result, patterns seen in such URLs can vary from those produced by people. Likewise, the authors of [39] and [40] suggested methods for identifying URLs produced by Domain Generation Algorithms (DGAs). A method for detecting fraudulent websites based on host-based and lexical information taken from URLs was suggested by the authors in [41]. The findings indicated that URL characteristics outperform the other feature categories in terms of accuracy.

In order to overcome the maximum sequence length (MSL) constraint in deep learning, the authors of [26] suggested an adaptive segmentation approach. utilising the Multi-Head

Self-Attention and multi-channel text convolution (MCTC) network, the detection model was built utilising the retrieved features from the webpage text, digital certificate, and Uniform Resource Locator (URL). Relying only on dynamic content characteristics, however, is difficult and may result in worse categorisation results. By modelling the stochastic system dynamics using deep Bayesian neural networks (DBNNs), the research in [42] offered a method for learning the uncertainty. To lower the dimensionality of the features vector, the authors in [43] introduced a feature extraction process called URL embedding based on an unsupervised learning technique called Huffman coding. The technique has been tested on a dataset that has significant assumptions about the length and distribution of the characters in the samples of malicious URLs, despite the fact that it performs better in detection than the current feature extraction algorithms.

An anomaly detection methodology for identifying malicious domains was put out by the authors in [34]. The normal profile of the normal domain was constructed using the Hidden Markov Model (HMM), which is a probabilistic model. Jensen-Shannon divergence between the suspicious domain and a subset of the benign domains is computed in the online process to determine if the domain is suspicious; if the JS divergence beyond a certain threshold, the malicious domain is identified. In order to increase the precision and effectiveness of detecting potentially dangerous web URLs, the authors of [31] devised a detection model named "deepBF" that combines Bloom Filters with Deep Learning algorithms. The detection classifier was built using an evolving convolutional neural network. In order to identify malicious URLs, the authors in [33] evaluate the effectiveness of several deep learning and conventional machine learning approaches. Among the classifiers under study, the BiLSTM classifier was said to perform the best. In order to enhance the learning process, the authors in [21] combined a number of feature

modifications to lower the amount of data. The solution included a variety of linear and non-linear spatial translation techniques. Even while feature modification greatly enhances classifiers built using conventional machine learning methods, the 62 features that were retrieved overall do not seem to be particularly difficult to classify using deep learning methods.

In order to address the rising worry of web-based assaults, the authors in [44] proposed a two-stage ensemble learning technique for malicious URL identification. To improve detection accuracy, the research makes use of cyber-threat intelligence characteristics from websites such as Whois and Google. Compared to conventional URL-based models, the two-stage ensemble technique, which combines Random Forest and Multi-Layer Perceptron algorithms, improves accuracy and lowers false positives. But the study doesn't fully look at the possible drawbacks of depending on outside cyber threat information sources, which might present problems with timeliness and comprehensiveness and call for further research.

Disadvantages

Due to complicated feature extraction, large data volumes, changing attack patterns, and the limits of conventional classifiers, existing systems often fail to identify malicious web apps using online content analysis. It is inadequate to rely just on lexical URL properties, which might result in incorrect classifications.

PROPOSED SYSTEM

To improve the identification of dangerous websites, the system suggests a unique multimodal representation technique that combines textual and image-based information. This method makes use of the advantages of both modalities: picture features identify more general hostile visual signals, while textual features record specific semantic information pertaining to assault patterns. Through picture analysis, hidden

patterns within written information may become apparent.

Two Convolutional Neural Networks (CNNs) are used in the suggested method: one for textual data and another for visual features. For better decision-making, their outputs are then pooled and fed into a classifier that uses artificial neural networks. Our findings show that the suggested model is better than the current methods. Our multimodal technique effectively detects malicious online apps, as shown by a 4.3% improvement in Matthews Correlation Coefficient (MCC) and a 1.5% decrease in the false-positive rate.

Advantages

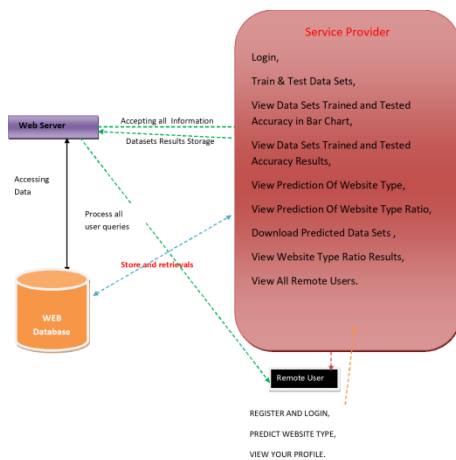
1. Malicious website identification becomes more thorough when DNS-derived characteristics are combined with URL-based ones. By providing useful contextual knowledge on domain behaviour and infrastructure, this synergy strengthens the assessment of website security and authenticity, leading to a more thorough and sophisticated method of spotting fraudulent websites.

2. The study introduces a multimodal representation approach that utilizes both textual and image-based features to represent a comprehensive feature set. While picture features are useful for identifying more generic harmful patterns, textual features help the deep learning model comprehend and convey specific semantic information pertaining to attack patterns.

3. To uncover hidden characteristics in the textual and visual representations, create two Convolutional Neural Network (CNN) models.

4. To discover the connections between the hidden characteristics that the CNN models were able to extract, a second deep learning classifier was built. By using deep learning methods to integrate and use both textual and visual information for more efficient malicious website identification, this method improves the field.

SYSTEM ARCHITECTURE



IV. IMPLEMENTATION

Modules

Service Provider

The Service Provider must use a working user name and password to log in to this module. Following a successful login, he may do several tasks including training and testing data sets, View Bar Charts of Trained and Tested Accuracy Data Sets View Accuracy Results for Trained and Tested Data Sets, View Website Type Prediction, View Website Type Ratio Prediction, Download Predicted Data Sets, View All Remote Users and Website Type Ratio Results.

View and Authorize Users

The administrator may see a list of all registered users in this module. Here, the administrator may see the user's information, like name, email, and address, and they can also grant the user permissions.

Remote User

A total of n users are present in this module. Before beginning any actions, the user needs register. Following registration, the user's information will be entered into the database. Following a successful registration, he must use his password and authorised user name to log in. Following a successful login, the user may do tasks including registering and logging in,

predicting the kind of website, and seeing their profile.

ALGORITHMS

Logistic regression Classifiers

The relationship between a collection of independent (explanatory) factors and a categorical dependent variable is examined using logistic regression analysis. When the dependent variable simply has two values, like 0 and 1 or Yes and No, the term logistic regression is used. When the dependent variable contains three or more distinct values, such as married, single, divorced, or widowed, the technique is sometimes referred to as multinomial logistic regression. While the dependent variable's data type differs from multiple regression's, the procedure's practical application is comparable.

When it comes to categorical-response variable analysis, logistic regression and discriminant analysis are competitors. Compared to discriminant analysis, many statisticians believe that logistic regression is more flexible and appropriate for modelling the majority of scenarios. This is due to the fact that, unlike discriminant analysis, logistic regression does not presume that the independent variables are regularly distributed.

Both binary and multinomial logistic regression are calculated by this software for both category and numerical independent variables. Along with the regression equation, it provides information on likelihood, deviance, odds ratios, confidence limits, and quality of fit. It does a thorough residual analysis that includes diagnostic residual plots and reports. In order to find the optimal regression model with the fewest independent variables, it might conduct an independent variable subset selection search. It offers ROC curves and confidence intervals on expected values to assist in identifying the

<https://doi.org/10.62643/ijerst.2025.v21.i2.pp420-431>

optimal classification cutoff point. By automatically identifying rows that are not utilised throughout the study, it enables you to confirm your findings.

Naïve Bayes

The supervised learning technique known as the "naive bayes approach" is predicated on the straightforward premise that the existence or lack of a certain class characteristic has no bearing on the existence or nonexistence of any other feature.

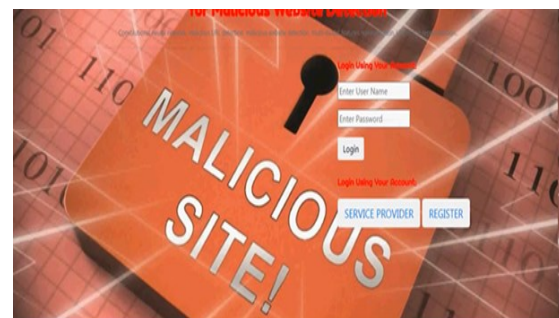
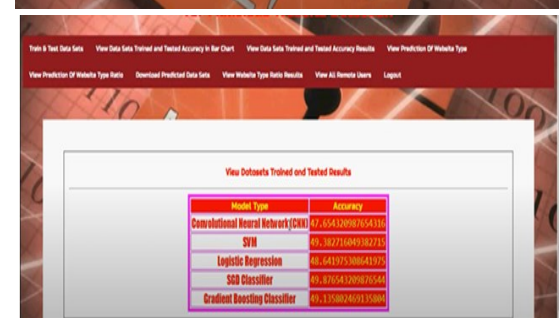
However, it seems sturdy and effective in spite of this. It performs similarly to other methods of guided learning. Numerous explanations have been put forward in the literature. We emphasise a representation bias-based explanation in this lesson. Along with logistic regression, linear discriminant analysis, and linear SVM (support vector machine), the naive bayes classifier is a linear classifier. The technique used to estimate the classifier's parameters (the learning bias) makes a difference.

Although the Naive Bayes classifier is commonly used in research, practitioners who want to get findings that are useful do not utilise it as often. On the one hand, the researchers discovered that it is very simple to build and apply, that estimating its parameters is simple, that learning occurs quickly even on extremely big datasets, and that, when compared to other methods, its accuracy is rather excellent. The end users, however, do not comprehend the value of such a strategy and do not get a model that is simple to read and implement.

As a consequence, we display the learning process's outcomes in a fresh way. Both the deployment and comprehension of the classifier are simplified. We discuss several theoretical facets of the naive bayes classifier in the first section of this lesson. Next, we use Tanagra to apply the method on a dataset. We contrast the outcomes (the model's parameters) with those from other linear techniques including

logistic regression, linear discriminant analysis, and linear support vector machines. We see that the outcomes are quite reliable. This helps to explain why the strategy performs well when compared to others. We employ a variety of tools (Weka 3.6.0, R 2.9.2, Knime 2.1.1, Orange 2.0b, and RapidMiner 4.6.0) on the same dataset in the second section. Above all, we make an effort to comprehend the outcomes.

V. SCREEN SHOTS

Model Type	Accuracy
Convolutional Neural Network (CNN)	87.65432987654321
SVM	89.10271684930271
Logistic Regression	88.64197530864197
SGB Classifier	89.47654329876543
Gradient Boosting Classifier	89.1358026013580



Website Prediction Type	Score
Malicious	95.33333333333333
Normal	96.66666666666667

VI. CONCLUSION

A malicious website detection model known as HF-CNN was created for this investigation. To improve the thoroughness of detecting dangerous

websites, the approach combines DNS and URL characteristics. To illustrate the merged feature set, a multimodal representation strategy that incorporates both textual and image-based attributes has been suggested. While picture attributes excel at identifying more general harmful patterns, textual attributes allow the deep learning model to understand and represent intricate semantic information related to attack patterns. To extract hidden features from the textual and visual representations, two Convolutional Neural Network (CNN) models were built. CNNs may record both local and global characteristics at the same time. The suggested model performs better than the other similar models, according to the findings. When compared to the baseline model, CNN, the overall performance in terms of F-measure and MCC has improved by 0.4% and 0.6%, respectively. There was a 1.6% decrease in the False Positive Rate (FPR) and a 1.4% decrease in the False Negative Rate (FNR).

The MMC score of 96.66% indicates that there are still significant mistakes in the detection performance, even if the suggested models had a high detection performance of 98.88% in terms of the F-measure. Unrepresented characteristics in DNS information and URLs were the main cause of the issues. Because certain innocuous domains with security flaws might become hostile via injection attacks, it is not a good idea to depend just on URLs, DNS information, or static properties for detecting rogue websites. As a result, integrating URL-based features with other features, such content features, is crucial. However, because of their high dynamic nature and capacity to be used by attackers to avoid detection, content characteristics are complicated. Therefore, further study is required to provide practical and efficient methods for obtaining online material.

Moreover, detection performance may be improved by using an adaptive ensemble of classifiers made to account for the dynamic nature of changing threats. Because each

classifier in the ensemble is built using a unique collection of characteristics, it may be used in a variety of threat situations with flexibility and resilience.

REFERENCES

- [1] NJ. (2023). How Many Websites are There in the World? Accessed: Sep. 10, 2023. [Online]. Available: <https://siteefy.com/how-many-websites-are-there/>
- [2] M. Liu, B. Zhang, W. Chen, and X. Zhang, "A survey of exploitation and detection methods of XSS vulnerabilities," *IEEE Access*, vol. 7, pp. 182004–182016, 2019.
- [3] J. Jang-Jaccard and S. Nepal, "A survey of emerging threats in cybersecurity," *J. Comput. Syst. Sci.*, vol. 80, no. 5, pp. 973–993, Aug. 2014, doi:10.1016/j.jcss.2014.02.005.
- [4] N. Provos, D. McNamee, P. Mavrommatis, K. Wang, and N. Modadugu, "The ghost in the browser: Analysis of web-based malware," *HotBots*, vol. 7, p. 4, Apr. 2007.
- [5] K. Townsend, "18.5 Million websites infected with malware at any time," *Wired Bus. Media, SecurityWeek*, Boston, MA, USA, Tech. Rep. Q4 2017, 2022. Accessed: Feb. 1, 2022. [Online]. Available: <https://www.securityweek.com/185-million-websites-infected-malware-any-time>
- [6] A. S. Raja, R. Vinodini, and A. Kavitha, "Lexical features based malicious URL detection using machine learning techniques," *Mater. Today*, vol. 47, pp. 163–166, Jan. 2021, doi: 10.1016/j.matpr.2021.04.041.
- [7] A. Subasi, M. Balfagih, Z. Balfagih, and K. Alfawwaz, "A comparative evaluation of ensemble classifiers for malicious webpage detection," *Proc. Comput. Sci.*, vol. 194, pp. 272–279, Jan. 2021, doi: 10.1016/j.procs.2021.10.082.
- [8] S. R. Zahra, M. A. Chishti, A. I. Baba, and F. Wu, "Detecting COVID-19 chaos driven phishing/malicious URL attacks by a fuzzy logic and datamining based intelligence system," *Egyptian Informat. J.*, vol. 23, no. 2, pp. 197–214, Jul. 2022, doi: 10.1016/j.eij.2021.12.003.

- [9] B. B. Gupta, K. Yadav, I. Razzak, K. Psannis, A. Castiglione, and X. Chang, "A novel approach for phishing URLs detection using lexical based machine learning in a real-time environment," *Comput. Commun.*, vol. 175, pp. 47–57, Jul. 2021, doi: 10.1016/j.comcom.2021.04.023.
- [10] R. Wazirali, R. Ahmad, and A. A.-K. Abu-Ein, "Sustaining accurate detection of phishing URLs using SDN and feature selection approaches," *Comput. Netw.*, vol. 201, Dec. 2021, Art. no. 108591, doi:10.1016/j.comnet.2021.108591.
- [11] D. K. Mondal, B. C. Singh, H. Hu, S. Biswas, Z. Alom, and M. A. Azim, "SeizeMaliciousURL: A novel learning approach to detect malicious URLs," *J. Inf. Secur. Appl.*, vol. 62, Nov. 2021, Art. no. 102967, doi: 10.1016/j.jisa.2021.102967.
- [12] K. Haynes, H. Shirazi, and I. Ray, "Lightweight URL-based phishing detection using natural language processing transformers for mobile devices," *Proc. Comput. Sci.*, vol. 191, pp. 127–134, Jan. 2021, doi: 10.1016/j.procs.2021.07.040.
- [13] S. Srinivasan, R. Vinayakumar, A. Arunachalam, M. Alazab, and K. Soman, "DURLD: Malicious URL detection using deep learning-based character level representations," in *Malware Analysis Using Artificial Intelligence and Deep Learning*. Berlin, Germany: Springer, 2021, pp. 535–554.
- [14] R. Chiramdasu, G. Srivastava, S. Bhattacharya, P. K. Reddy, and T. Redd Gadekallu, "Malicious URL detection using logistic regression," in *Proc. IEEE Int. Conf. Omni-Layer Intell. Syst. (COINS)*, Aug. 2021, pp. 1–6, doi: 10.1109/COINS51742.2021.9524269.
- [15] N. M. Phung and M. Mimura, "Detection of malicious Javascript on an imbalanced dataset," *Internet Things*, vol. 13, Mar. 2021, Art. no. 100357, doi: 10.1016/j.iot.2021.100357.
- [16] Y. Huang, T. Li, L. Zhang, B. Li, and X. Liu, "JSContana: Malicious Javascript detection using adaptable context analysis and key feature

extraction,” *Comput. Secur.*, vol. 104, May 2021, Art. no. 102218, doi:10.1016/j.cose.2021.102218.

[17] R. Rakesh, S. Muthurajkumar, L. SaiRamesh, M. Vijayalakshmi, and A. Kannan, “Detection of URL based attacks using reduced feature set and modified C4.5 algorithm,” *Adv. Natural Appl. Sci.*, vol. 9, no. 6, pp. 304–311, 2015.

[18] S. Kim, J. Kim, and B. B. Kang, “Malicious URL protection based on attackers’ habitual behavioral analysis,” *Comput. Secur.*, vol. 77, pp. 790–806, Aug. 2018, doi: 10.1016/j.cose.2018.01.013.

[19] S. He, B. Li, H. Peng, J. Xin, and E. Zhang, “An effective cost-sensitive XGBoost method for malicious URLs detection in imbalanced dataset,” *IEEE Access*, vol. 9, pp. 93089–93096, 2021.

[20] D. R. Patil and J. B. Patil, “Malicious URLs detection using decision tree classifiers and majority voting technique,” *Cybern. Inf. Technol.*, vol. 18, no. 1, pp. 11–29, Mar. 2018.

[21] T. Li, G. Kou, and Y. Peng, “Improving malicious URLs detection via feature engineering: Linear and nonlinear space transformation methods,” *Inf. Syst.*, vol. 91, Jul. 2020, Art. no. 101494, doi: 10.1016/j.is.2020.101494.

[22] S. Wang, Z. Chen, Q. Yan, K. Ji, L. Peng, B. Yang, and M. Conti, “Deep and broad URL feature mining for Android malware detection,” *Inf. Sci.*, vol. 513, pp. 600–613, Mar. 2020, doi: 10.1016/j.ins.2019.11.008.

[23] R. Vinayakumar, K. P. Soman, and P. Poornachandran, “Evaluating deep learning approaches to characterize and classify malicious URL’s,” *J. Intell. Fuzzy Syst.*, vol. 34, no. 3, pp. 1333–1343, Mar. 2018, doi:10.3233/JIFS-169429.

[24] B. Liang, M. Su, W. You, W. Shi, and G. Yang, “Cracking classifiers for evasion: A case study on the Google’s phishing pages filter,” in *Proc. 25th Int. Conf. World Wide Web*, Apr. 2016, pp. 345–356.

[25] Y. Shi, G. Chen, and J. Li, “Malicious domain name detection based on extreme machine learning,” *Neural Process. Lett.*, vol. 48, no. 3, pp. 1347–1357, Dec. 2018.