JCBI
Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 09 Issue 01
2025

Research Article
https://doi.org/10.56979/901/2025

Webpage Classification for Search Engine Optimization using Machine Learning

Khurram Zeshan Haider[1], Rimsha Zafar[1], Qamas Gul Khan Safi[2*], Muhammad Awais[1], and Muhammad Munwar Iqbal[2]

[1]Department of Software Engineering, Government College University, Faisalabad, Pakistan.
[2]Department of Computer Science, University of Engineering and Technology Taxila, 47050, Taxila, Pakistan.
[*]Corresponding Author: Qamas Gul Khan Safi. Email: qamas.gul@uettaxila.edu.pk

Received: March 13, 2025 **Accepted:** May 06, 2025

Abstract: Webpage classification for SEO is an essential area of study where machine learning, especially Deep Neural Networks (DNNs), plays a crucial role. This paper aims to develop an accurate Malicious & Benign page classifier using Deep Neural Networks (DNNs) for webpage classification in SEO. Data collection, selecting features, model construction, training, and evaluation, handling data that is imbalanced, & practical implementation considerations are just a few of the elements that make up the research approach. This dataset contains features like raw webpage content, geographical location, JavaScript length, obfuscated JavaScript code of the webpage, etc. The dataset has about 1.5 million web pages. 300,000 are used for testing, while 1.2 million are used for training. This dataset is highly skewed as 98.35% of the dataset are Benign webpages, and 2.27% are Malicious webpages, with a training dataset totaling 40,1806 instances, consisting of 25,770 good webpages, 6.41%, and 9472 harmful webpages, 2.35%. Our model is trained rigorously to identify patterns indicative of malicious intent. Our algorithm demonstrates robustness in classification in a test dataset of 398125 instances, including 23298 good webpages 5.8% and 9344 harmful webpages (2.34%). So, choosing the evaluation metrics carefully is essential, as just accuracy won't give the correct evaluation, so I use an F1-score of 97.73%, a recall score of 95.2%, a precision score of 96%, and a confusion matrix. As a result, this paper solves the challenge of accurately differentiating between malicious and benign websites. The outcomes of this research contribute to webpage classification in SEO by leveraging DNNs to accurately classify malicious and benign webpages.

Keywords: Malicious & Benign Websites; Machine Learning; Deep Neural Network; URL; SEO (Search Engine Optimization)

1. Introduction

Webpage classification is crucial for effective search engine optimization (SEO), and Deep Neural Networks (DNNs), in particular, are machine learning methods that have shown promise in achieving accurate classification. This paper focuses on developing an accurate Malicious & Benign Webpage Classifier using DNNs for SEO. The two classes represent malicious and good websites, respectively. A similar disparity can be seen in our dataset because there are more benign websites than malicious websites on the internet. The majority of webpages are benign, as can be seen from the graphical representation in Class Labels.

Deep Neural Network has been used for developing a classifier to differentiate malicious webpages from normal ones. Another purpose of using DNN is to achieve better search engine optimization, safety, and usability. The steps to follow are dataset assembly, selection of feature set, training, testing, and evaluation. For this work HTML structure, nature of the content, and URL properties were also considered. After preprocessing, supervised learning with the option of the backpropagation mechanism was followed

for better model exhibition. The aim is to improve webpage classification for search engine optimization (SEO) so that it can promptly handle dynamic content by focusing on helpful features in a customized fashion. Two classes, termed as Malicious and Benign, are concluded for webpages by using deep Neural Networks.

Webpage classification for SEO by machine learning entails employing Deep Neural Networks (DNNs) to differentiate between malicious and benign web pages. Using protection systems based on the windows' orientation, placement, and features is one of the primary methods to reduce energy consumption and prevent the risk of overheating [2]. Our primary goals are to accurately identify unknown information and perform factor evaluation of the website categories to classify benign websites and others that help users avoid the risk of malicious websites. The probability is calculated using Naïve Bayes and other valuable techniques to evaluate and modify the website categorization model.

2. Literature Review

In this section, current research has been explored in the areas of webpage classification using deep neural networks and other machine learning techniques, specifically focusing on developing a Malicious & Benign page classifier using DNNs. Our findings demonstrate that the benign and different test responses compare favorably to each other, indicating that the Naïve Bayes method, which has an accuracy of up to 90%, is better suited to handle distinguishing high-quality websites [3]. Because of Naïve Bayes' strong impact, harmful JavaScript code is becoming increasingly common on websites. JavaScript vulnerabilities are used by many of the recent assaults, and obfuscation is sometimes used to mask the malicious intent and prevent discovery. Creating a sufficient intrusion-detection system (IDS) for malicious JavaScript to protect Internet users is necessary—a fresh set of capabilities that recognize JavaScript obfuscation. The features have been chosen based on the identification of obfuscation, a common technique to evade traditional malware detection systems. The effectiveness of the proposed method has been evaluated using attacks that obfuscate JavaScript [4]. Previous research has distinguished between different file formats, redirected URLs, and other methods to identify these malicious domains. The primary effect of a malicious host is to either send the user malicious files or data, or it will redirect the user to unwanted files or data. Performance for DNNSFLO-based host recognition is compared with support vector machines (SVM), fuzzy C-means algorithm (FCM), fuzzy k-nearest neighbors (FKNN), neural networks (NN), artificial neural networks (ANN), and Naïve Bayes. Python is used to implement this host-detecting procedure. [5] Web spam tactics frequently make it difficult to identify malicious sites and cause inefficiencies. Next, we took the viewpoint of the end-user. We utilized CNN to learn & recognize website screenshot photos to address the issues brought on by spam techniques or to improve the detection efficiency. [6] Classifiers were constructed and trained to categorize an unknown sample (a web page) into one of the three predefined classes & to determine the key elements influencing the degree of page adjustment. The suggested method has practical significance in that it offers the foundation for developing software agents or expert systems that automatically identify webpages or sections of webpages that require modification to adhere to SEO best practices and, consequently, possibly receive higher search engine rankings. Additionally, this study's findings advance the subject of determining the ideal values for the ranking variables that search engines employ to assign a page's rating. This involves creating a feature vector encapsulating characteristics like URL length, geographic location, top-level domain, and JavaScript content [7]. The dataset is expanded using ISO Code 3 as a feature, which improves the classifier's performance. The classifier can identify geographic differences in web content by linking each webpage to its corresponding ISO 3166-1 alpha-3 code [8]. When evaluating the position of each nation in the network, centrality & similarity are complementary. For example, centrality is primarily determined by the number of links between countries, while the caliber of digital trade chapters determines similarity. According to our interpretation, a nation with a high level of centrality in the network will probably be a rule maker, whereas a nation with a high level of similarity will probably be a rule promoter. DNNs, or deep neural networks) Between a neural network's input and output layers, a deep neural network, also known as a DNN, comprises numerous hidden layers of units. DNNs can represent complex nonlinear relationships [9]. The multinomial NB classifier assumes that the attributes derive from multinomial distributions. This version, often used in NLP use cases like spam classification, helps use discrete data, including frequency counts. The Naive Bayes classifier is applied when dealing with binary

variables with only two values, true or false, 1 and 0. This specific version is known as Bernoulli Naive Bayes (Bernoulli NB) [10].

Training the DNN involves iteratively adjusting internal parameters to minimize the disparity between predicted and actual labels. We approached distinguishing between benign and harmful websites as an issue of classification. We used this data to apply several machine-learning techniques, including logistic regression, random forests, decision trees, and support vector machines. Several assessment criteria, including false positive rate, accuracy, precision, recall, and F1 score, were employed to gauge how well each categorization method performed. The machine learning techniques become biased during training towards a particular class of websites because the dataset was unbalanced [11]. Malware, spam, and phishing are examples of cyberattacks. Websites are the most used Internet component; hence, hackers find them to target for attacks. To improve security, both individuals and organizations must identify harmful websites. An acceptable solution to accurately distinguish between benign and dangerous websites. We suggest using machine learning techniques to categorize websites as benign or harmful based on their URL and other host-based characteristics [12]. However, a new direction for unsafe website detection could be pursued with distinct feature representations and preprocessing techniques [13]. To some extent, traditional techniques like blocklisting and signature-based techniques can be used as first checks before implementing learning-based models. The model's efficacy is gauged using classification reports and confusion matrices. The trained DNN model can be saved for future deployment, streamlining the classification of webpages and enhancing SEO techniques [14].

Supervise learning uses data with labels to train machine learning models. In supervised learning, where both the input & the outcome are known in advance, the model learns using labelled data. Known cyber risks are frequently associated with specific IP Addresses from which malicious websites frequently originate. A classifier can identify & flag websites with potentially hazardous material to strengthen cybersecurity efforts by analyzing IP Addresses. The objective is to develop a mapping function that, using what it has learnt from the labelled examples, can accurately predict labels for brand-new, unused data. The data preprocessing techniques influence the model's training and classification impact. Since every data preparation technique has its own set of restrictions, several preprocessing techniques exist for various datasets [15]. One categorization task is intrusion detection. We primarily employ one-hot coding, numerical techniques, and three standardized processing methods for the network's traffic dataset before integrating it into the DNN model. They are introduced individually below. According to the observation that hostile resources are frequently similar to one another, for example, because they develop using common attack toolkits, there is an alternative method to looking for dangerous content [16]. The classification approaches of Supervised Learning using Machine Learning (ML) assess different supervised learning algorithms and identify the most effective algorithm based on variables (features), data set, and number of instances [17]. According to the findings, SVM was determined to be an algorithm with the highest accuracy and precision. Accordingly, it was discovered that the next most accurate classification algorithms after SVM were Naïve Bayes and Random Forest [10]. Next, using a separation plane, the SVM is utilized to build a model to determine whether a new sample instance falls into one of two categories. Systems that use this categorization technique can provide small sample sets with enhanced learning capabilities. The SVM technique can benefit applications like facial recognition, web page identification, and network intrusion detection. High training or decision rates, indifference to input data dimension, or continuous correction of numerous parameters with increased training data, which improves the system's capacity for self-learning, are advantages of SVM in intrusion detection systems [18]. 30% of this dataset was used for testing and validation, while the remaining seventy percent was used to train models. Eight essential input data points were used in the analysis: The Internet Protocol (IP) address, the site's HTTPS status, the number of dots within the domain name, and a few top-level domain-related features. The URL, IP address, code in JavaScript, obfuscation code, top-level domain, country, and HTTPS were among the many characteristics considered for the classification. The training set had a class imbalance, with most of the samples being benign and the minority being malicious [19]. IP addresses significantly boost the classifier's capabilities when classifying webpages for SEO by machine learning approaches, particularly when creating a deep neural network (DNN) based Malicious & Benign Webpage Classifier. IP addresses on the internet can be categorized into different network types, including Class A, Class B, and Class C networks. Code injection's primary building components are HTML and JavaScript code. Attackers can

gain personal information or execute malicious code to install malware on a victim's computer by exploiting weaknesses in HTML and JavaScript scripts [20].

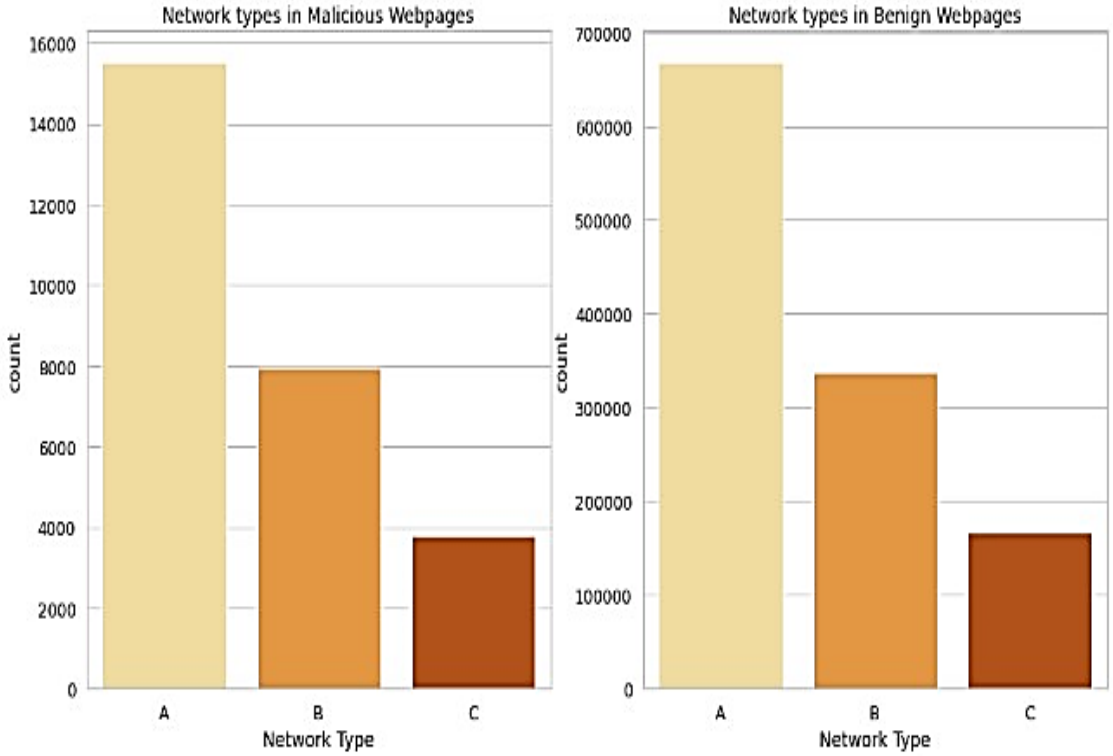


Figure 1. Network type dependents i.e., network's class, purpose, and technology requirements.

Developing a deep neural network (DNN)-based Malicious or Benign page classification focuses heavily on JavaScript, Special Characters, Content, Links, HTTPS, HTTP, and Labels.

These factors are also crucial in the research area of webpage categorization for SEO. These components are regarded as features and properties inside the dataset in the study process for creating a malicious or benign webpage classifier [21]. They give the DNN-based classifier helpful context and information, enabling it to decide to categorize webpages. By including these traits, the classifier improves SEO techniques by spotting essential elements that affect webpage security and ranking, thus advancing SEO methods and techniques. This manuscript's dataset uses machine learning-based analysis of benign and harmful web pages. It facilitates learning that is both supervised and unsupervised. The Kaggle platform added class labels for benign and malicious web pages to the dataset for supervised learning. This dataset already contains the most essential features that were extracted within the scope [22].

Table 1. Webpage features or attributes that provide information about webpages.

Sr. No.	url_ len	geo_l ec	tld	who_is	htt ps	js_l en	js_obf _len	label
0	28	United States	com	complete	yes	106.0	0.0	good
1	41	United States	ac.jp	complete	yes	144.0	0.0	good
2	53	Japan	org	complete	yes	102.5	0.0	good
3	54	United States	edu	incomplet e	no	13.45	0.0	good
4	19	Ukraine	com	complete	yes	0.0	0.0	good
…	…	……	… ….	…… ….	… …..	… …..	……..	… ….
9995	46	Australia	edu	complete	yes	65.0	0.0	good
9996	27	South Africa	com	complete	yes	148.0	0.0	good
9997	42	Colombia	com	complete	no	106.0	0.0	good
9998	39	Japan	co.uk	complete	yes	34.0	0.0	good

9999	50	China	org	complete	yes	42.5	0.0	good

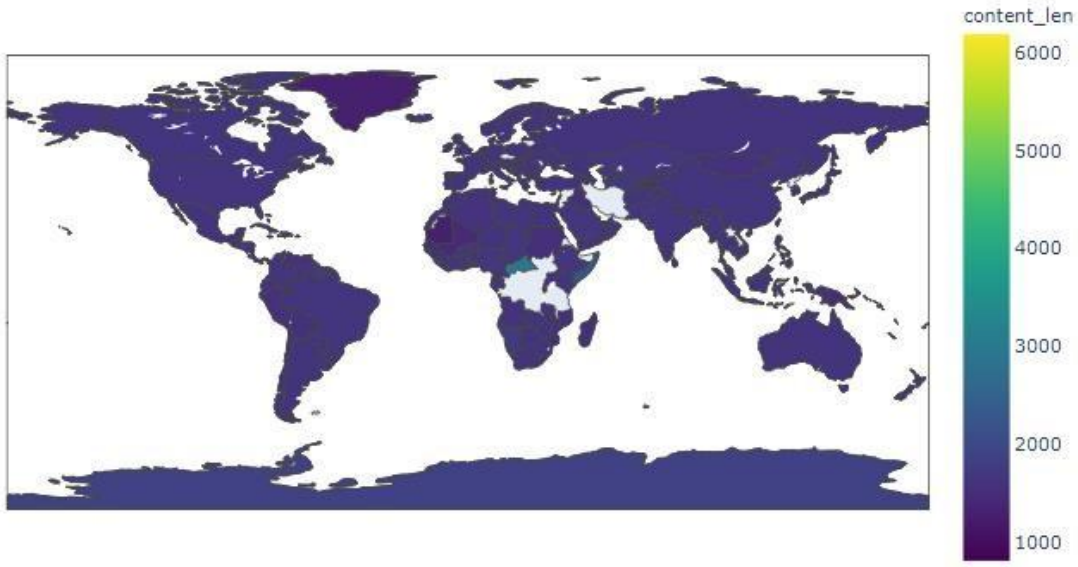


Figure 2. Webpage Feature Map for SEO Classification (Malicious vs. Benign)

These webpage features provide a comprehensive set of attributes that can be used to train machine learning models for webpage classification [23]. Scholars employ diverse methodologies incorporating unique characteristics to identify malevolent websites within cyberspace. Most of them take essential elements from URL text to classify dangerous web pages because of their ease of use and risk-free character [24]. By analyzing these features, the model can identify patterns, anomalies, and characteristics distinguishing malicious and benign web pages and make informed decisions to improve SEO strategies and web security [25]. The malicious server is easily identified, and any disruption to cloud services is immediately prevented. Therefore, this network's efficiency naturally rises. It involves figuring out how to categorize various malicious

Host types using a feature-based method. Malicious URLs link users to additional malicious and phishing web pages or to sites where attackers can install malware. The input data is a list of source URLs. They will be included in a list of locations to browse after being searched for sub-URLs. Every vulnerability has a detection method developed for it. Every vulnerability has a unique set of URL filtering attached. These filters classify and condense input data URLs according to data from the fields, forms, characteristics, and variables users use in the application. An HTTP request & HTTP response are used in the automated framework test method [26]. Harmful attack web detection software that can identify C&C-containing domains, waypoints, malicious websites, and malicious code distribution sites. We also wish to investigate a sub-model that uses webpage binary imaging and deep learning to determine whether a web page is dangerous [27]. Evaluation measures are covered in detail in this section. First, we define terminology like false-negative (FN), false-positive (FP), true-negative (TN), and true-positive (TP). The total amount of malicious domains that were accurately classified as malicious is shown by TP. The number of benign domains that were successfully found is indicated by TN. The quantity of FP represents benign domains mistakenly classified as malicious, whereas the A malicious domain's count is indicated with FN [28]. They, therefore, tried to classify URLs using less expensive sets of features before acquiring the more costly ones. With a score of 90.51%, the experimental results demonstrated that RF performed better than any other ML method in terms of accuracy & time. Nevertheless, a model for identifying fraudulent URLs was presented, and it performed better (93.30%) using the same classifier. They separated the dataset in ratios of 60:40, 70:30, & 80:20 between training and test data. For every split ratio, they computed the accuracy across multiple iterations. They concluded that the ideal split was an 80:20 split [29]. Using the extracted data, the suggested method was classified using a hybrid Deep Neural Network-Shuffled Frog Leap Optimization (DNN-SFLO) classifier. The experimental findings using the suggested framework demonstrate high classification rates utilizing machine learning algorithms. According to further research findings, executable file information is quite good at differentiating between harmful and benign software [30].

3. Proposed Methodology

The categorization of websites is an essential step in the software industry. Making accurate estimates before the project starts will significantly affect how successfully it is performed. Therefore, for web categorization on any dataset, a model that can minimize error magnitude is needed. I have selected three publicly available datasets. In the next step, the data preprocessing is performed on all the datasets and the features of the variables in all the datasets. Furthermore, the null values were checked in all three datasets. The next face is the use of different models. Three ML algorithms are applied in this study: DNNs, Naive Bayes, and 3rd algorithm, SOM, on three datasets. The next step is using an Evaluation matrix to check the models' performance. For checking the model's performance, we have used the three evaluation matrices, which are F1 Score, Recall, and Precision. If we obtain a suitable output, the algorithms stop; otherwise, it goes back to the dataset collection step. The data preprocessing step is explained in detail below.

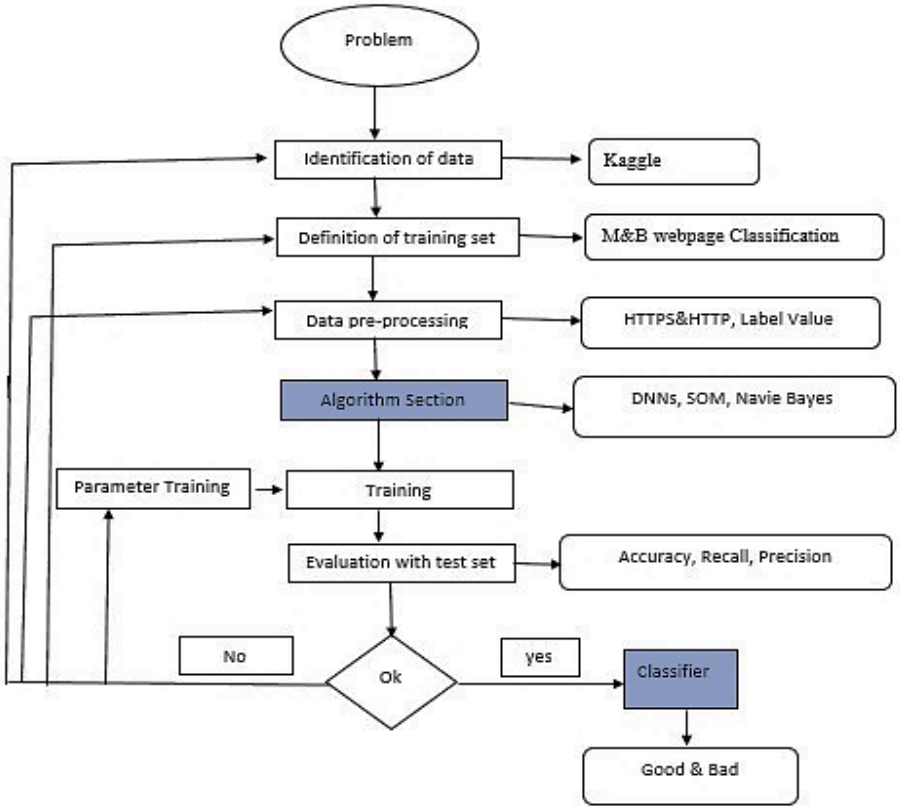


Figure 3. Conceptual Framework of Malicious and Benign Webpage Classification for SEO

3.1. Dataset Report

3.1.1. Malicious and Benign Webpage Classification

These datasets are used for the classification of websites' Malicious and Benign webpage classification, the SOM model of the webpage, and a page classification model with ML and ML for webpage content classification. There are 11 features and 10000 instances in the malicious webpage classification datasets. There are eight features and 10000 cases in the SOM dataset. There are two units of target features (Good/ yes) label or (Bad/No) label for webpage classification datasets.

The dataset has around 1.5 million web pages total, with 1.2 million used for testing and 300,000 for training. This table's features include the dataset's raw webpage content, Location, JavaScript code length, and the webpage's obfuscated JavaScript code. Using Google's Safe Browsing API, the labels were verified and confirmed. Malicious websites download malware onto your computer, causing it to be an issue, gather your data, and many more worst-case scenarios.

The final results of our investigation into the top domains associated with malicious web pages. For search engine optimization (SEO) purposes, our research uses machine learning approaches to categorize websites as dangerous or benign. Top-level domains (TLDs) with a history, such as .com, .org, and .net, are frequently used by malicious websites to make their websites appear legitimate. DNN models

effectively classified malicious webpages based on domain features, enhancing SEO security, and the final results of our investigation into the top domains associated with benign webpages. For search engine optimization (SEO) purposes, our analysis uses machine learning approaches to categorize websites as dangerous or harmless.

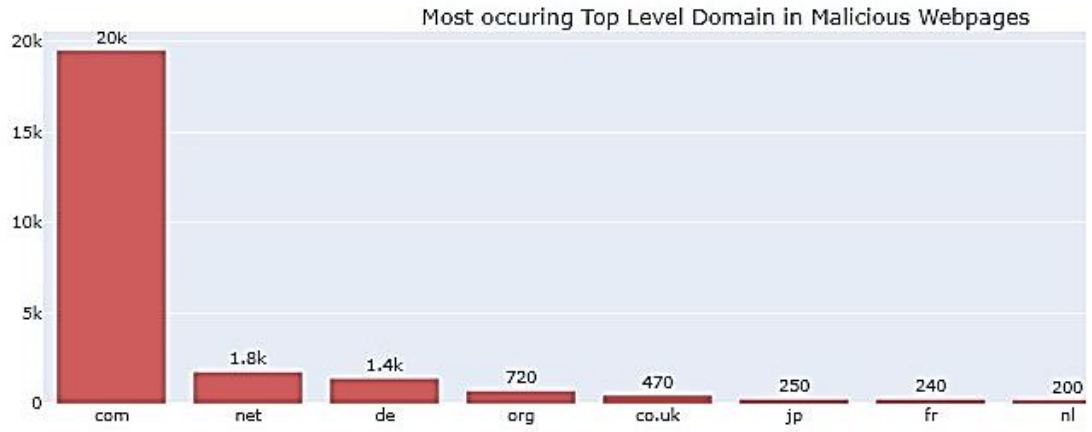


Figure 4. DNN models Top domains as malicious & benign web page features, enhancing SEO security.

3.1.2. SOM Model for Webpage Classifier

A Self-Organizing Map (SOM) applied to webpage data, utilizing features like URL, label, JavaScript length, obfuscated JavaScript length, HTTPS usage, TID (Transaction ID), WHOIS information, and geographical location data, allows for clustering and visualizing webpages. This aids in identifying patterns, grouping similar web content, and enhancing data analysis for tasks such as content classification and anomaly detection.

3.1.3. DNN Model for Webpage Classification

The dataset was split into training, validation, and test sets. The training set is used to train the model, the validation set helps tune hyperparameters, and the test set evaluates the model's performance. Fine-tune hyperparameters like learning rate, batch size, and the number of hidden layers or neurons using the validation dataset to achieve the best performance.

When building a Deep Neural Network (DNN) model for webpage classification, the architecture typically includes various layers, each with different types, output shapes, and parameters. The specific architecture will depend on your data and problem, but I can provide a basic design of a feedforward neural network for webpage classification.

4. Results and Discussion

Table 2. DNN webpage classification

Layer (type)	Output Shape	Param #
keras_layer (KerasLayer)	(None, 20)	400020
dense (Dense)	(None, 128)	2688
dense_1 (Dense)	(None, 64)	8256
dropout (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 32)	2080
dense_3 (Dense)	(None, 16)	528
dense_1 (Dropout)	(None, 16)	0
dense_4 (Dense)	(None, 1)	17

Total params: 413, 589; **Trainable params:** 413,589 ; **Non_Trainable params:**0

The number of parameters (or "Param #") in each layer can be calculated as described above. Remember the exact architecture, including the number of layers and neurons. These layers are used for learning patterns in the data. The output shape depends on the number of neurons in the layer. A correlation heat map of numerical attributes is a powerful visualization tool that can provide insights into the relationships between different numerical features in webpage classification for SEO, distinguishing between malicious and benign web pages. A positive correlation (e.g., a correlation coefficient close to 1) indicates that the other also tends to increase as one attribute increases. A negative correlation (e.g., a correlation coefficient close to 1) suggests that the other tends to decrease as one attribute increases. Correlation coefficients near zero indicate that the attributes have little to no linear relationship.

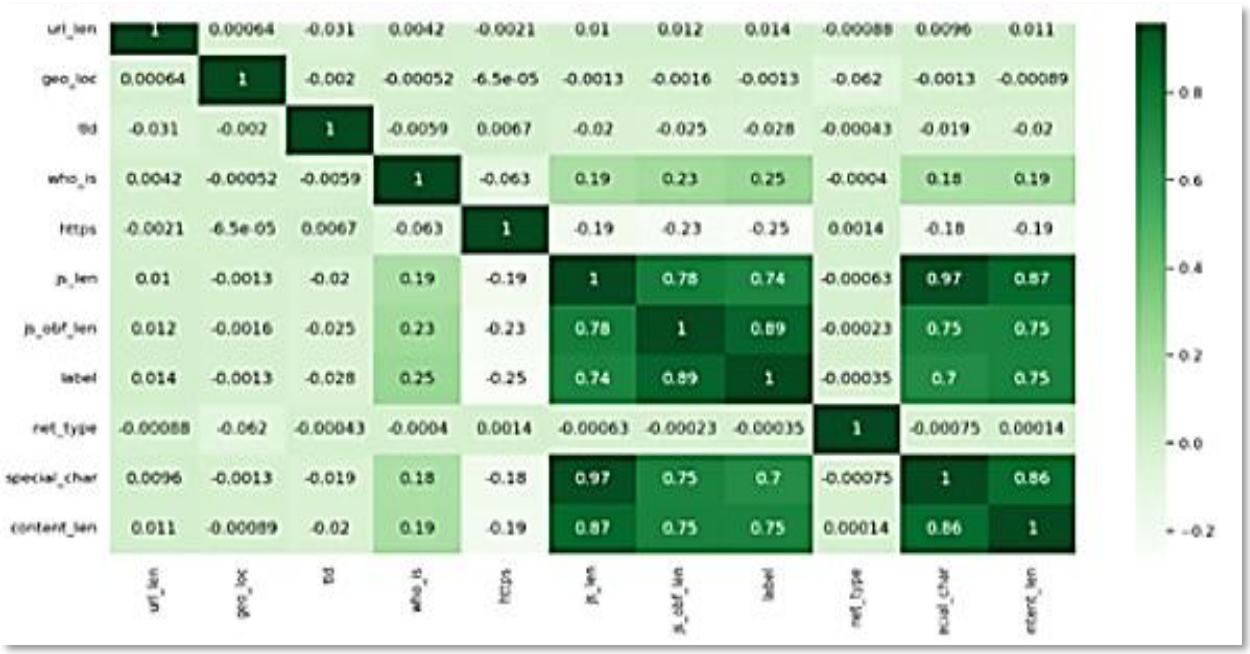


Figure 5. Correlation Features map of Numerical Attributes

Pearson Correlation: The most often used correlation metric, correlation, explains the linear relationship between two variables results from the Pearson correlation coefficient analysis range from -0 to 1. When the value between two features is zero, there is a negative correlation between the two values. However, if the amount is 1, the two properties perfectly correlate. A strong association is indicated if the resulting values range from -0 to 1. Correlation coefficients in statistics determine how closely both variables are related.

Evaluation Matrix: I used the f1_score, recall, and confusion matrix as our assessment metrics because the dataset is significantly skewed, with benign websites making up 97.302% and malicious web pages making up 2.27%.

Confusion Matrix: Some are bad (negative), while others are good (benign). I want to accurately educate computers on how to distinguish between them. When they do something well (true positives & negatives), I can determine it, and when they do something incorrectly (false positives & false negatives). This enables us to choose SEO strategies more wisely. To evaluate how well the robots I create work, I use a variety of metrics, such as precision, recall, accuracy, and the F1 score.

In this paper, we apply machine learning approaches to analyze the global reach of harmful websites, and we provide the results. Malicious websites are a common problem that can be found all over the world. My investigation gave to important information about where dangerous websites are located.

The application of Self-Organizing Maps (SOM) enabled us to:

Cluster Malicious Webpages: SOM revealed clusters of similar malicious webpages, aiding in categorizing and understanding these threats.

- Enhance Classification: SOM-based analysis improved the accuracy of our classification system.
- DNN-Based Analysis
- Deep Neural Network (DNN) approach offered:

- High Classification Accuracy: DNN demonstrated its effectiveness in accurately classifying malicious and benign web pages.
- Efficiency: DNN-based models provided efficient real-time protection against malicious content.

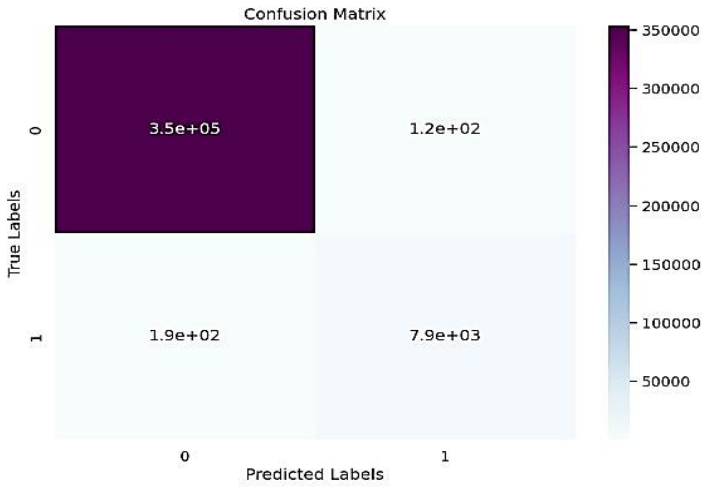


Figure 6. Malicious& Benign webpage classification of Confusion Matrix Results

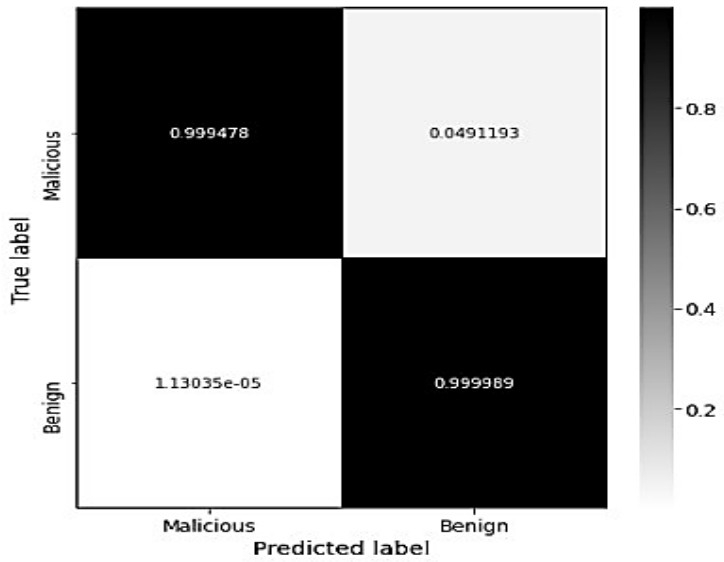


Figure 7. SOM Confusion Matrix for webpage classification

Figure 8. IP address geographic distribution: malicious

This section presents the paper's final results and insights, focusing on the global distribution of benign web pages using machine learning techniques.

Figure 9. IP address geographic distribution: benign

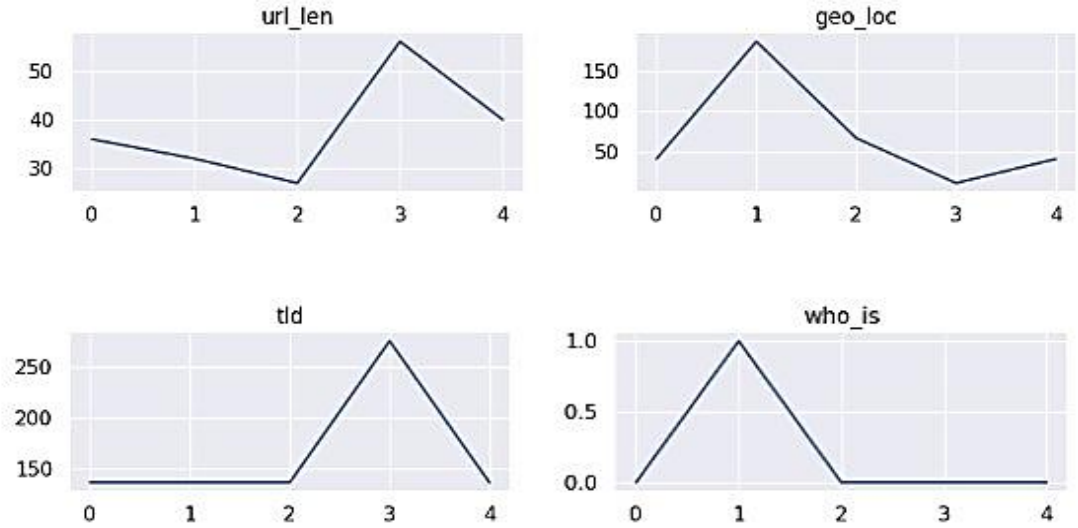


Figure 10. Values of features for Malicious& Benign webpage classification

The length of a URL is the count of characters or bytes. Web developers, content creators, and SEO engineers use URL length for achieving better user understanding and improved usability.

SOM-Based Analysis: If Self-Organizing Maps (SOM) were employed, here's what we discovered:

- **Content Clusters:** SOM helped identify clusters of benign webpages with similar content, facilitating content categorization.
- **Localized SEO:** The insights from SOM support tailoring SEO strategies to specific regions.

DNN-Based Analysis:

- **Precise Classification:** DNN models effectively classified benign webpages, aiding in SEO optimization.
- **Real-Time SEO:** DNN-based systems offer real-time assistance for SEO professionals in maintaining high search engine rankings.

Research provides insights into the global distribution of benign webpages, offering implications for SEO professionals, content creators, and users worldwide.

URL features like length, structure, and specific keywords might provide information about the nature of a webpage. Features like URL, geo_location (geo_lec), content length (content_len), top-level domain (TLD), and WHOIS information (who_is) were gathered and prepared for analysis.

For URL length distribution the Kernel Density Estimate (KDE) plot shows concentration for Malicious while for Benign multiple peaks indicates diverse URL lengths. Self-Organizing Maps (SOM) and Deep Neural Networks (DNN) were used for comparative analysis where our focused application is SEO.

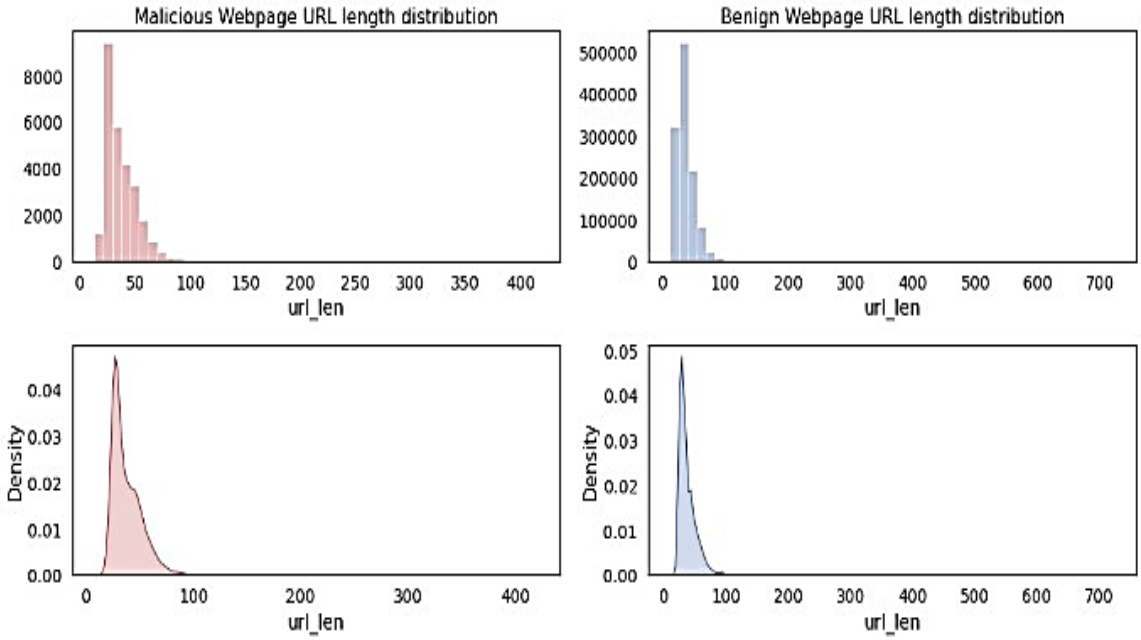


Figure 11. Malicious and Benign URL length distribution

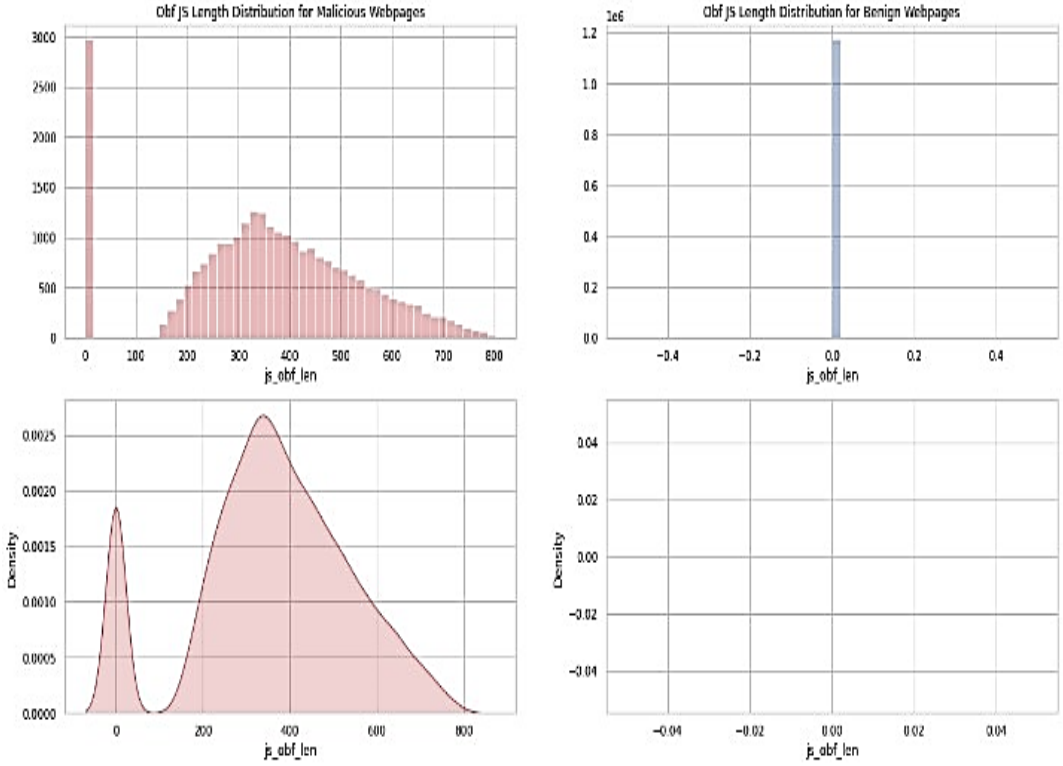


Figure 12. JS length distribution for both malicious and benign webpages

Broader and more visualized distribution is obtained using Violin plot for Benign webpages. A specific length pattern appears for URLs in SOM. It has vital effect in search engine page ranking and ultimately the usability.

Mean Content-Length of Malicious Webpage: 6926.395993101677

Mean Content-Length of Benign Webpage: 1519.0496117235857

The study shows JavaScripte JS has proved an important feature and its length directly influence SEO rankings. Bar chart in the figure depict characteristics or attributes differences between the two classes i.e. malicious and benign.

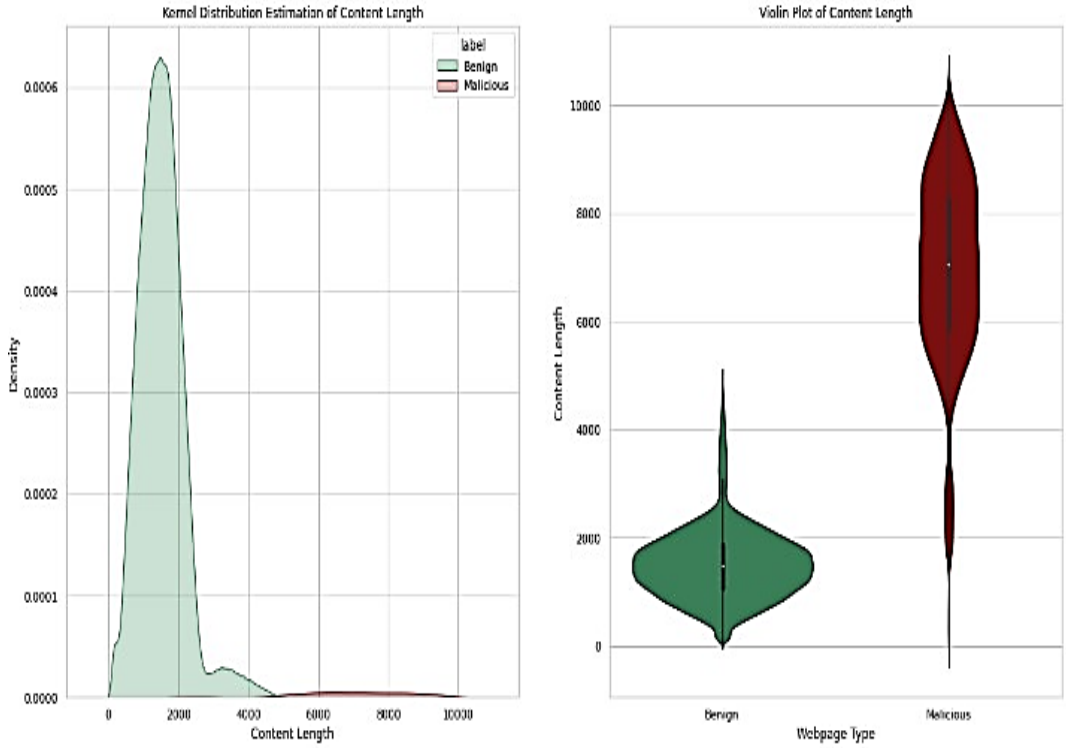
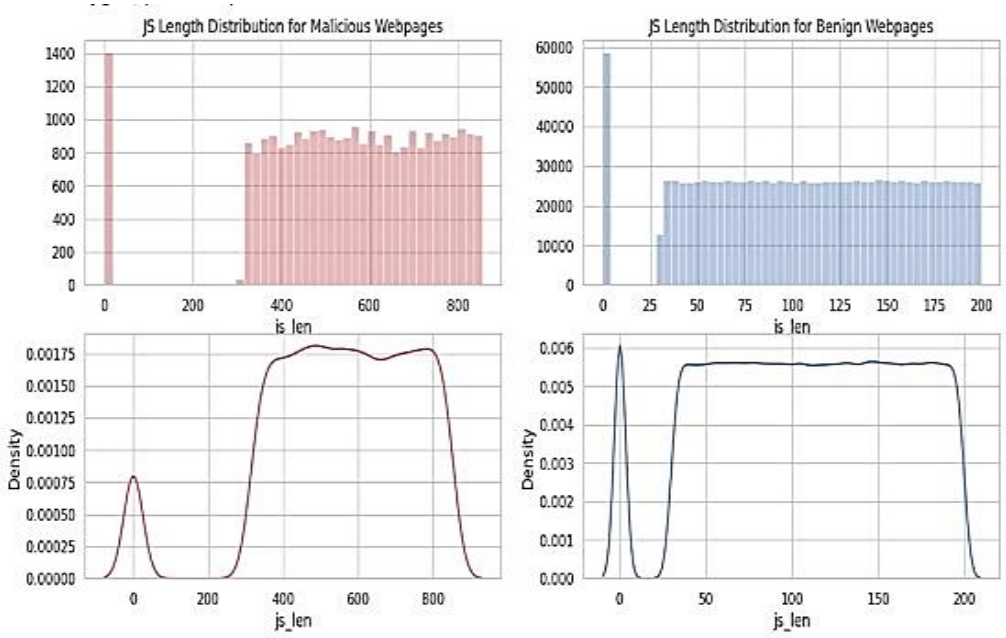


Figure 13. Kernel Distribution Estimation and Violin Plot of the length of content.

In this study upon using DNN it was found that the classification based on JavaScript length count exhibits high accuracy i.e. 99.73%. As shown in the testing phase on a dataset for 1.2 malicious or 1.2 benign data



.

Figure 14. JS Length distribution for webpage classification

Obfuscated URL length has also emerged as a critical feature when it is been exposed to DNN when a self-organizing map SOM was sketched. The essential role of obfuscated URL length in webpage classification for SEO provides valuable guidance for SEO strategies and content optimization.

js Violin plot: The violin plot for malicious webpages reveals a relatively concentrated distribution of obfuscated JS lengths. The violin plots depicting obfuscated JS length distribution clearly represent how obfuscation impacts webpage classification in SEO. These insights can inform SEO strategies, content optimization, and cybersecurity efforts, ultimately enhancing search engine rankings and online security.

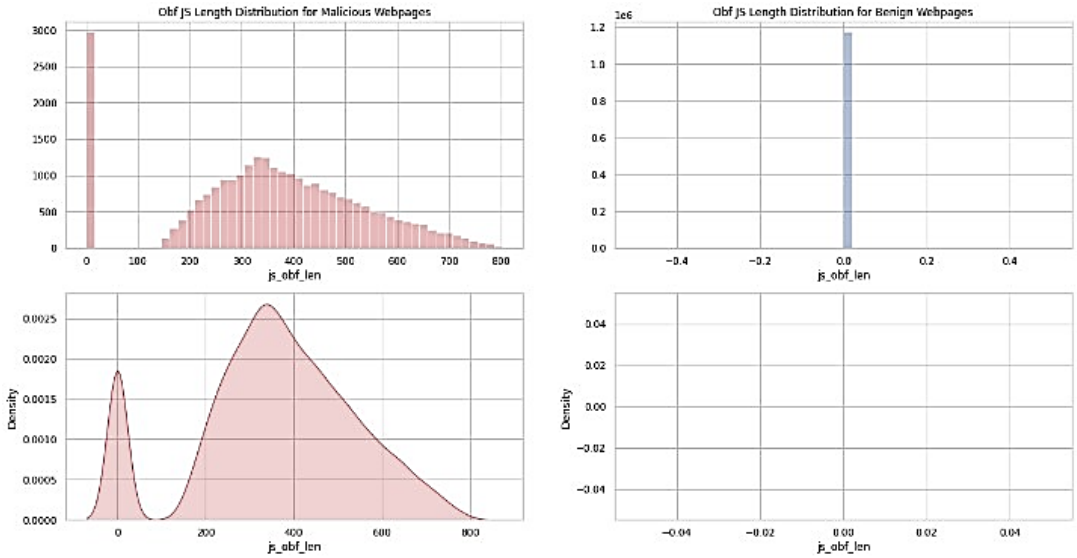


Figure 15. Obfuscated JavaScript Distribution for Malicious and Benign webpage

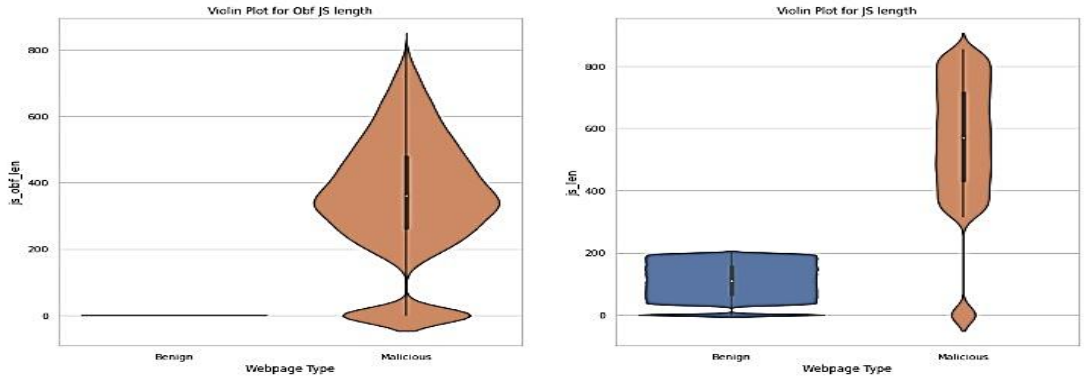


Figure 16. JS Violin plot of malicious & Benign Webpage Classification

The average content length of webpages worldwide, mainly for webpage classification for SEO (Search Engine Optimization) and detecting malicious and benign webpages using machine learning, can be challenging due to the vast diversity of web content.

Table 3. Average Content length of the webpages around the world

iso_3	content_len
ABW	1386.176471
AFG	1648.366197
AGO	1636.500000
AIA	1343.500000
ALB	1616.111111

As for global averages, these can fluctuate over time and are influenced by regional and cultural factors. Studies or reports on global or industry specific averages can provide more accurate and up-to-date data on webpage content lengths. Content length is one of many potential features to consider in these tasks. Moreover, the ideal content length for SEO can differ based on a website's specific goals and audience.

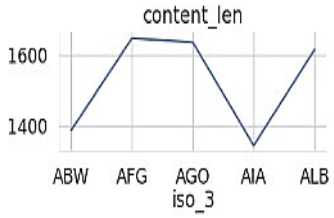


Figure 17. Value of ISO3 webpage classification for SEO

For the average content length of webpages worldwide or webpage classification for SEO and security, you would need to collect data, conduct analysis, and implement machine learning models independently of ISO3 codes.

Time Series:

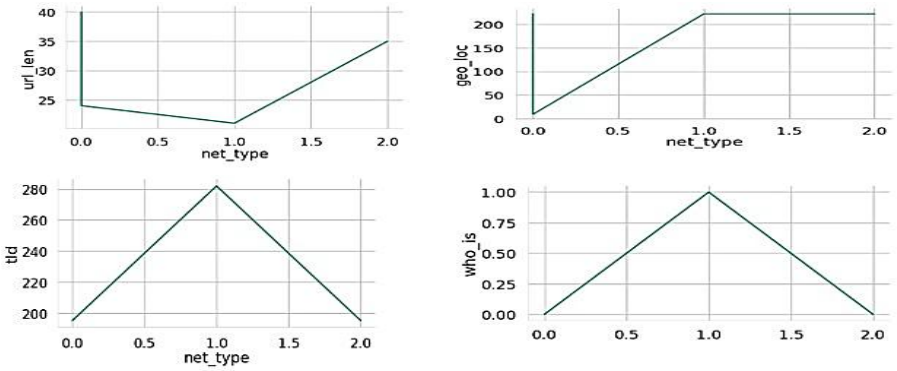


Figure 18. Time-space graph webpage classification for SEO

Creating time series graphs for webpage classification involves visualizing how webpages are categorized or labelled over time. This can be useful for tracking trends, monitoring changes, and understanding the evolution of webpage classification results. A line chart is often used to visualize changes in webpage classification labels over time. Each line represents a specific classification (e.g., benign, malicious), and the x-axis represents time. The y-axis indicates the count or percentage of web pages falling into each category.

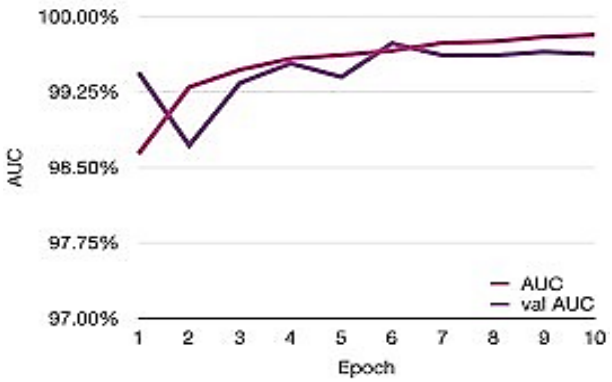


Figure 19. Time series evolution of webpage classification results

Two class classification of malicious and benign focusing SEO at webpage level, the importance was given to the fact of achieving balance between high malicious detection count and false positive minimization.

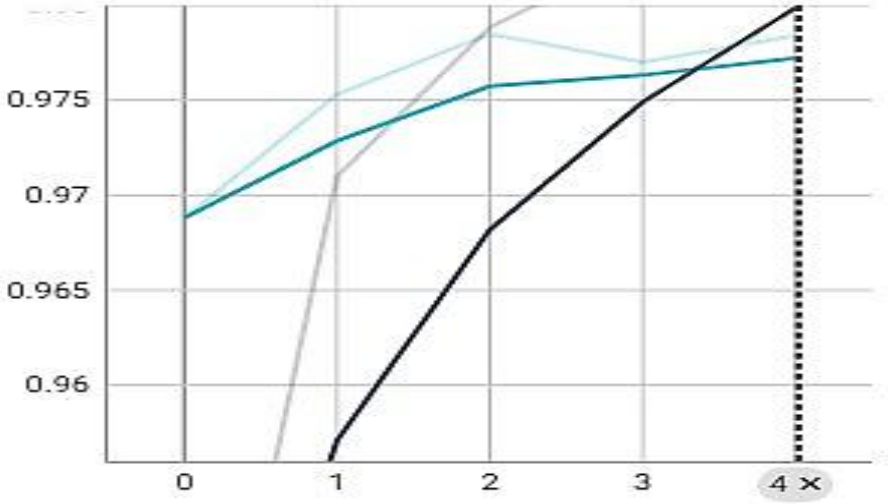


Figure 20. Plotting of Metrics: Loss, AUC, Precision & Recall of webpage classification models

The graph shows wider area under the curve > .97 reflects higher depicted difference between studied classes. Low loss shows better model performance and it is precise like it possesses less count of also positives. The plot shows higher recall value which promises fewer false negatives.

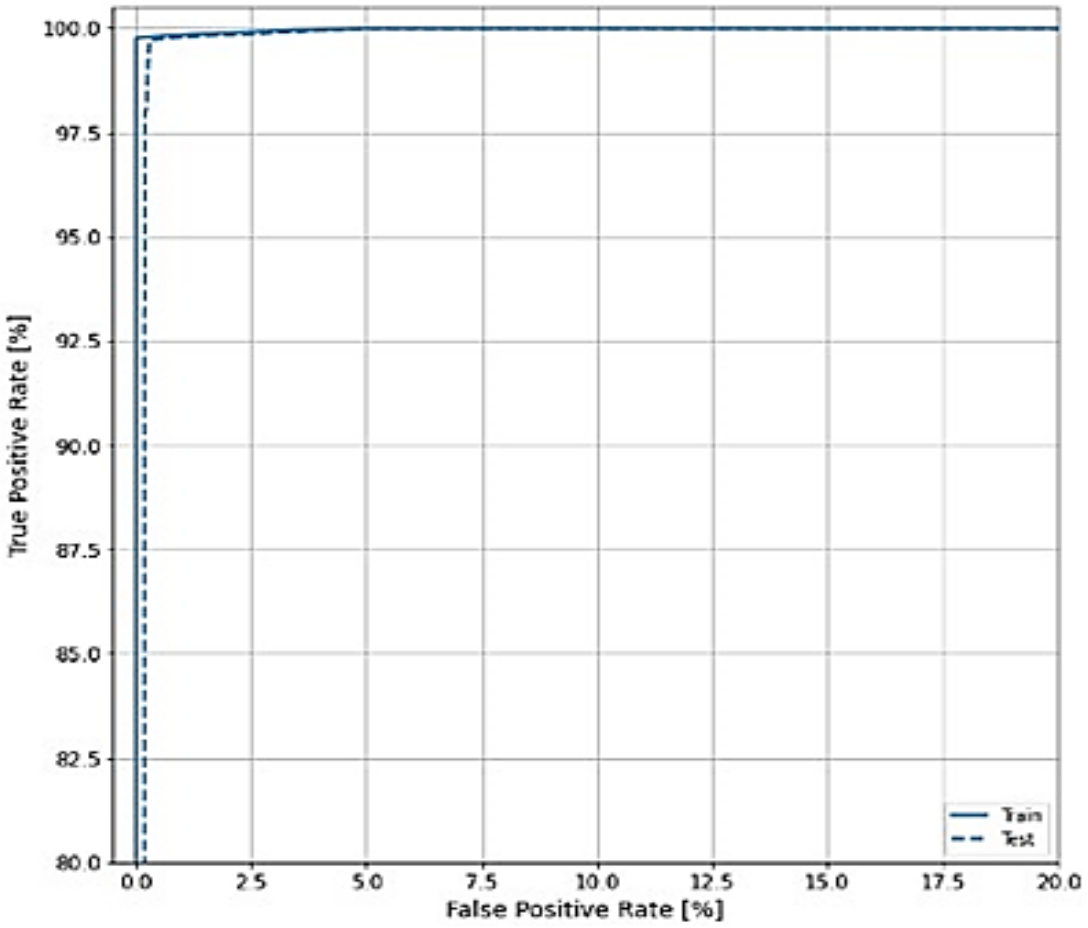


Figure 21. AUC and Epoch time for webpage classification

- The diagonal line from (0,0) to (1,1) represents the performance of a random classifier.
- The curve is closer to the upper-left corner indicates a better-performing classifier.

Loss is a numerical value quantifying how well the model's predictions align with the proper labels. Various loss functions (e.g., cross entropy, mean squared error) can be used based on the nature of the classification problem. In the evaluation phase, the loss assesses the model's performance on a validation or test dataset. Lower loss values indicate better model performance by monitoring the loss and adjusting the number of epochs, machine learning model for accurate and robust classification of malicious and benign webpages.

In SEO webpage classification, the number of epochs is an important hyper-parameter to consider during model training & testing. The goal is to balance training and testing the model long enough to learn the underlying patterns but not so long that it starts overfitting. By monitoring the AUC and considering the number of epochs, you can optimize the machine learning model for accurate and robust classification of malicious and benign webpages for SEO purposes.

It helps us understand how well the model distinguishes between the two classes and decides the trade-off between actual positive rate (sensitivity) and false positive rate (1-specificity). The valid positive rate represents the proportion of true positive cases (malicious web pages) the model correctly identifies as positive. The false positive rate is the proportion of real negative cases (benign webpages) the model incorrectly identifies as positive (malicious). The ROC plot represents the actual positive rate (sensitivity) against the false positive rate (1-specificity) at various threshold settings. The curve typically starts at the point (0,0) and ends at (1,1). A point in the upper-left corner (0,1) represents a perfect classifier with a high actual positive rate and no false positives.

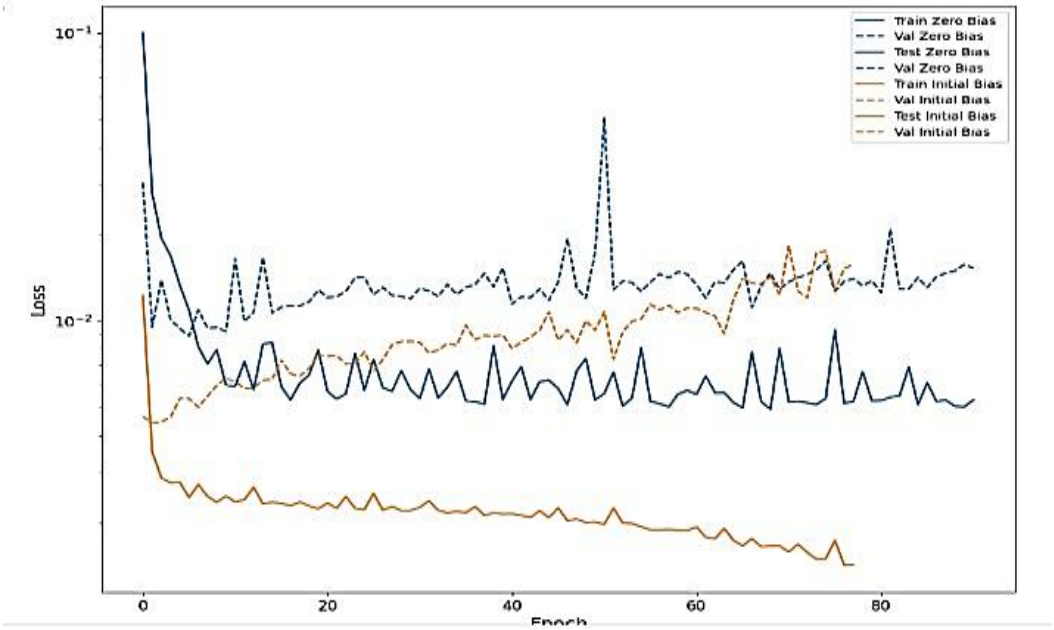


Figure 22. Time-space graph webpage classification for SEO by machine learning of Influence of Initial Bias

Table 4. The final results of a critical investigation into the influence of initial bias

	Malicious	**Benign**
Malicious	0.999478	0.049119
Benign	0.000011	0.999989

The study utilizes advanced machine learning techniques, specifically Self-Organizing Maps (SOM) and Deep Neural Networks (DNN), to assess initial bias affecting these classifiers' performance in distinguishing between malicious and benign webpages. The analysis demonstrates how variations in initial bias affect the SOM classifier's ability to organize and classify web pages. It identifies optimal initial bias settings for maximizing classification accuracy.

Accessing Accuracy of Estimates*:* F1 Score: The F1 score determines how accurate the model is, which computes the calculated average between the two different metrics and considers both precision and recall. It uses the harmonic mean instead of the arithmetic mean to assess recall and precision simultaneously.

$$F1 = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$$

Recall: Classifying them as positive determines the proportion of real positives that were caught. Recall is crucial for evaluating model performance, just like accuracy and precision. It is calculable that use the values given in the confusion matrix to calculate the following equation:

$$\text{Recall} = \frac{TP}{TP+FN}$$

Precision: Precision performance indicator assesses the applicability of a model's output. The values in the confusion matrix can be used in the equation that follows to calculate it:

$$\text{Precision} = \frac{TP}{TP+FP}$$

Table 5. Accuracy Report: 99.938% Report of Classification:

	precision	**recall**	**f1score**	**support**
Class 0	1.00	1.00	1.00	353872
Class 1	0.99	0.98	0.99	8062
accuracy			1.00	361934

macro avg	1.00	0.99	0.99	361934
weighted avg	1.00	1.00	1.00	361934

Where 0 represents the Benign while 1 is for the malicious class.

Our research differentiates between benign and malicious web pages with a training dataset totaling 40, 1806 instances, consisting of 25,770 Benign webpages (6.41%) and 9472 malicious webpages (2.35%). Our model is trained rigorously to identify patterns indicative of malicious intent.

The algorithm proves to be robust as we had total 398125 instances of test portion of the dataset, which includes Benign Webpages 23,298 (5.8%) and of malicious ones 9,344 (2.34%). So, we have used data-driven approach to achieve visibility, robustness and security.

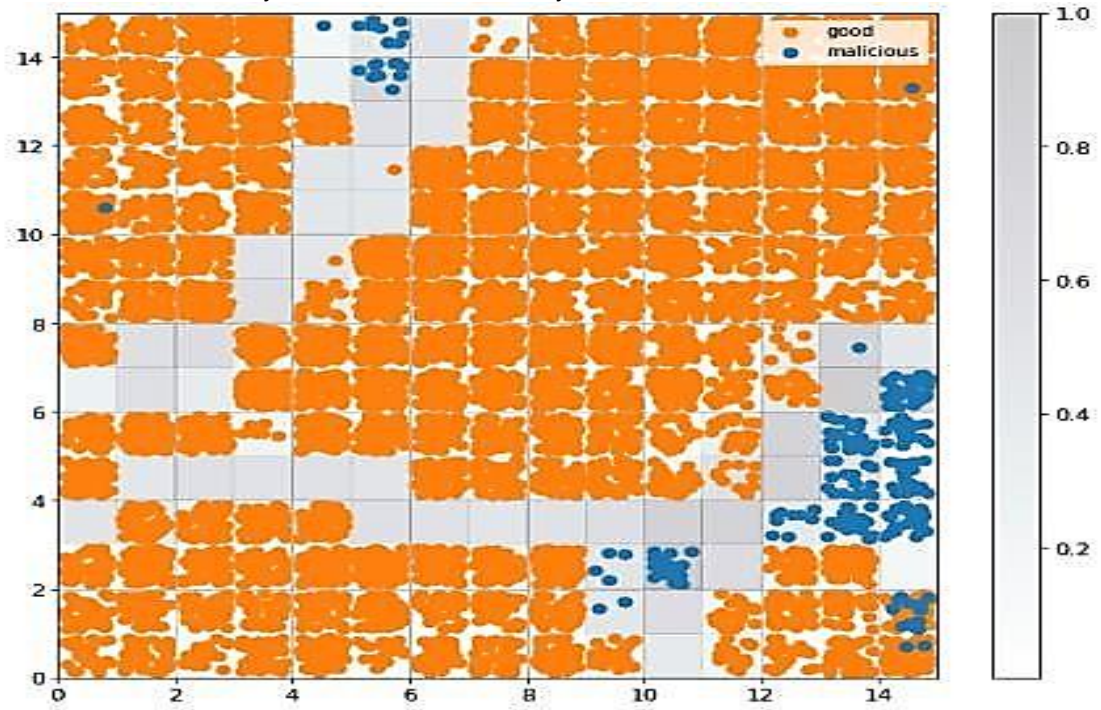


Figure 23. Displaying the Malicious and Benign Webpage Classification Scatter Plot of Labels

5. Conclusion and Future Work

I have explored the realm of webpage classification for SEO using advanced machine learning techniques, focusing on Deep Neural Networks (DNNs), SOM, and Naïve Bayes. The primary objective was to create a robust classifier that effectively distinguishes between malicious and benign web pages. To assess classifiers' effectiveness, I employed a range of metrics, including accuracy 99.738%, precision [(0,1), (1,0.99)], recall [(0,1), (1,0.98)], and the F1-score [(0,1), (1,0.99)]. Using machine learning, the SOM Model is the array value 1,0 for webpage classification SEO. I also addressed the challenge of imbalanced data by employing oversampling and under sampling techniques to enhance classification accuracy. The collection includes features taken directly from websites and can be used to categorize websites as harmful or benign. The dataset also contains the raw page content, geo_lec, tId, HTTPS, url_len, som_wt, label, and js_obf_len, including JavaScript code that can be utilized for machine learning purposes or to obtain more properties. Beyond the technical considerations, I talked about our models' actual use, scalability, and possible interaction with existing SEO procedures. The future work of malicious and benign webpage classification for SEO using machine learning holds tremendous innovation potential. By following these methods, I can continue to improve the accuracy and reliability of webpage classification systems, ultimately benefiting SEO professionals and internet users worldwide.

References

1. Ganesh, V., et al., Phishing Website Detection Using Machine Learning. 2023.
2. Norouziasas, A., et al., Implementation of ISO/DIS 52016-3 for adaptive façades: A case study of an office building. Building and Environment, 2023. 235: p. 110195.
3. Wang, S., Y. Wang, and M. Tang. Auto Malicious Websites Classification Based on Naive Bayes Classifier. in 2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE). 2020. IEEE.
4. Alazab, A., et al., Detection of obfuscated malicious JavaScript code. Future Internet, 2022. 14(8): p. 217.
5. Nandita, G. and T. Munesh Chandra, Malicious host detection and classification in cloud forensics with DNN and SFLO approaches. International Journal of System Assurance Engineering and Management, 2021: p. 1-13.
6. Liu, D. and J.-H. Lee, CNN-based malicious website detection by invalidating multiple web spams. IEEE Access, 2020. 8: p. 97258-97266.
7. Matošević, G., J. Dobša, and D. Mladenić, Using machine learning for web page classification in search engine optimization. Future Internet, 2021. 13(1): p. 9.
8. Wang, X., et al., Binary classification of welding defect based on deep learning. Science and Technology of Welding and Joining, 2022. 27(6): p. 407-417.
9. Lee, K.Y., et al., Network and Text Analysis on Digital Trade Agreements. KIEP Research Paper, World Economy Brief (WEB), 2023: p. 23-03.
10. Osisanwo, F., et al., Supervised machine learning algorithms: classification and comparison. International Journal of Computer Trends and Technology (IJCTT), 2017. 48(3): p. 128-138.
11. Ul Hassan, I., et al., Significance of machine learning for Detection of malicious websites on an unbalanced dataset. Digital, 2022. 2(4): p. 501-519.
12. Kaddoura, S. Classification of malicious and benign websites by network features using supervised machine learning algorithms. In 2021 5th Cyber Security in Networking Conference (CSNet). 2021. IEEE.
13. Lavreniuk, M. and O. Novikov, Malicious and benign websites classification using machine learning methods. Theoretical and Applied
14. Cybersecurity, 2020. 2(1).
15. Sen, M., K.S. Ray, and A. Chakrabarti. Malicious URL Classification Using Deep Neural Network. in 2021 IEEE 18th India Council International Conference (INDICON). 2021. IEEE.
16. Ritwika, S. and K.B. Raju. Malicious Software Detection and Analyzation Using the Various Machine Learning Algorithms. in 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT). 2022. IEEE.
17. Gao, M. et al., Malicious network traffic detection based on deep neural networks and association analysis. Sensors, 2020. 20(5): p. 1452.
18. Invernizzi, L., et al. Evilseed: A guided approach to finding malicious web pages. In 2012 IEEE Symposium on Security and Privacy. 2012. IEEE.
19. Mohan, P.V., et al., Leveraging computational intelligence techniques for defensive deception: a review, recent advances, open problems and future directions. Sensors, 2022. 22(6): p. 2194.
20. Abad, S., H. Gholamy, and M. Aslani, Classification of Malicious URLs Using Machine Learning. Sensors, 2023. 23(18): p. 7760.
21. Raja, A.S., et al. Malicious Webpage Classification Based on Web Content Features Using Machine Learning and Deep Learning. in 2022 International Conference on Green Energy, Computing and Sustainable Technology (COST). 2022. IEEE.

22. Maurya, S. and A. Jain. Malicious Website Detection Based on URL Classification: A Comparative Analysis. in Proceedings of Third International Conference on Computing, Communications, and Cyber-Security: IC4S 2021. 2022. Springer.
23. Singh, A., Malicious and benign webpages dataset. Data in brief, 2020. 32: p. 106304.
24. Claudia-ioana, C., Malicious Web Links Detection-A Comparative Analysis of Machine Learning Algorithms. Studia Universitatis Babes-Bolyai, Informatica, 2023. 68(1).
25. Pradeepa, G. and R. Devi. Web Content Based Features for Malicious Web Page Detection Using Machine Learning. in 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA). 2022. IEEE.
26. Nagaonkar, A.R. and U.L. Kulkarni. I am finding the malicious URLs using search engines. In 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom). 2016. IEEE.
27. Saxena, A., et al. Detection of web attacks using machine learning-based URL classification techniques. In 2022 2nd International Conference on Intelligent Technologies (CONIT). 2022. IEEE.
28. Shin, S.-S., S.-G. Ji, and S.-S. Hong, A Heterogeneous Machine Learning Ensemble Framework for Malicious Webpage Detection. Applied Sciences, 2022. 12(23): p. 12070.
29. Priya, S., V.D. Reddy, and V. Balaji Using AI to detect and classify malicious domain names.
30. Shaffi, S.S. and I. Muthulakshmi. Search Engine Optimization using Machine Learning for Web Page Classification. in 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS). 2022. IEEE.
31. Kamboj, A., et al., Detection of malware in downloaded files using various machine learning models. Egyptian Informatics Journal, 2023. 24(1): p. 8194.