Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Machine Learning-Powered DNS Firewall

Asma Ahmed A. Mohammed

Department of Computer Science, University of Tabuk, Tabuk, Saudi Arabia

With the quick development in landscape of cybersecurity, the importance of DNS firewall solutions has been recently pronounced. Such solutions work as building blocks in forming inoficial access to various domains, suggesting real-time protection and gretaly unclear communications. The conventional paradigm depends heavily on preprepared lists of known malicious domains, necessitating frequent updates to maintain relevance. However, this method shows inadequate in yet-to-be-cataloged malicious or domains identifying emerging, leading to potential vulnerabilities. Throughout this paper, a creative research endeavor is discussed to shed lights on presenting a cutting-edge DNS firewall solution that proves the power of Machine Learning (ML) techniques. The major purpose is to use the real-time detection of malicious domain requests, thereby critically enhancing cybersecurity protocols. A reasonable assembled dataset, incorporating 34 intricate features and meticulously recorded instances totaling 90,000, was critically chosen from genuine DNS logs. Similarly, it becomes more riched through the careful integration of Open-Source Intelligence (OSINT) sources. The set goal includes the empowerment of precise in addition to rapid classification of domain requests as either malicious or benign.
Keywords: c DNS, Machine learning, Deep learning, Cybersecurity, XGBoost, SVM, Random Forest, LightGBM.

1. Introduction

The obvious enhancement of the network has called out and introduced the way to companies as well as entrepreneurs to develop on emerging prospects and engage their process as well as evolution. The development in the advancements of technology has ushered in a concomitant surge in critical risks, especially those connected to the increasing frequency of network security. This paradigm change has created an environment where many enterprises discover that they are threatened by serious problem from malicious

presentors seeking unreal and unethical access to sensitive data, as it is happening in the recent instances [1]. Based on this view and in relation to evolving security landscape, organizations are obliges to include robust countermeasures as well as strengthening their cyber defenses to relief the unauthorized inherent in the contemporary digital milieu. In the domain of network dynamics, it is crucial to announce that the increased intricacy coming from the extensive and rapid growth of the network infrastructure has concurrently simplify the emergence of discernible lacunae, thus allowing new and opportunistic assailants with avenues to incorporate in prohibitive activities. A harmful attempt like the above mentioned one, if left unsolved, has the abilit to ruin and destroy the confidentiality of different entities such as transgressing the governmental secrecy as well as manipulating the personal privacy of everyday users [2].

This considers the crucial significance of measures and comprehensive strategies that are proactive to enhance the network architecture in the face of danger, building a well secured and flexible digital ecosystem. In spite of the industrious presence of security services by companies to decrease possible threats, assailants precisly utilize strategies that varies in order to control these measures and secretly get the access to the network. It is important to note that one of the considerable exploited methods by these adversaries include the manipulation of the Domain Name System (DNS) protocol, working as a conduit to access restricted data. Some studies have shown the seriousness of this issue, maintaining that a staggering 87% of companies fell prey to DNS attacks in the year 2021 [3]. This highlights the real challenges that can be found in cybersecurity efforts, prioritizing a continual enhancement and reassessment of defensive protocols to achieve the tactics created by professional threat actors.

As a matter of fact, the Domain Name System (DNS) is considered to be a functional part of the pivotal internet protocol, framed by the significant responsibility of identifying and naming computer resources accessible through Internet protocols (IP). DNS efficiently directs user requests to the ultimate hosting machin which is operating as an intricate system, facilitate the identification of the sought-after resources manifested in the form of Uniform Resource Locators (URLs) containing the associated domain name [4]. At its main part, the crucial aim of the DNS system is to translate these user-friendly URLs into IP addresses, giving a format which is more comprehensive and yet memorable for users to navigate the huge expanse of the digital realm. Such transformation attempts to reduce an issue which is cognitive and is associated with struggling with an integration of arbitrary letters and numbers, similarly enhancing the act of being moe accessible and usable of the online resources [5]. The hidden and real goal of the DNS underscores lies in its role in making the internet more user-friendly, including an easy, understandable, and healthy communication not only between users but also on the countless collection of computer resources available on the Internet. Because DNS as an ideas is not connected to data transfer, in theory it should not be a problem for companies or firms. Regrettably, what is mention can't announce and mean that these companies are too safe from DNS-related threats.

The Domain Name System (DNS) unfolds as systematic process through the operational framework. Based on the user interaction with a website, including activities such as browsing or search queries, the Domain Name System (DNS) system specifically

encompasses communication with the nearest root name server. Such interaction is initiated to solicit responses pertaining to the user's inquiries, referred to as the "requested query." Building on this, the root server forms connections with Top-Level Domain (TLD) servers, systematically collating the integral components of the domain name to determine the comprehensive IP address associated with the designated website. Noteworthy TLDs include extensions such as '.com', '.org', and '.edu'. It is quite worthy to mention that local DNS serves has an effective role in enabling Internet Service Providers (ISPs) to monitor internet traffic [6].

Moreover, there exists the inherent possibility for domains to adopt a malevolent disposition within the intricate domain of Domain Name System (DNS), functioning as conduits for the propagation of malware, facilitation of Command and Control (C&C) communications, and hosting of phishing or spam websites [7]. Both the domain name and subdomain name form the nefarious domains which lead to a serious threat vector that can precipitate internet attacks. When utilized to construct malicious Uniform Resource Locators (URLs), the malevolent nature of these domains becomes more clear. The uniform resource locator (URL) is made up, converging to constitute the host name. This host name function as a representative identifier for the computing entity that hosts the pertinent internet resource. Through the intricacies of these irrelated processes, the importance of vigilant cybersecurity measure to counteract the potential exploitation of DNS vulnerabilities in the ever-evolving landscape of online threats is shown.

The historical landscape of many domain detection initially relied on conventional methods and strategy, including the examination of web content, URL inspection, and the analysis of network traffic. However, as technological sophistication advanced, a pivotal transformation transpired, ensuring the integration of artificial intelligence (AI) for the automated identification of malicious domains [8]. This contemporary era allows the application of advanced AI methods, getting away from traditional strategies and allowing0 a more proactive and adaptive stance.

This progress in the side of malicious domain detection commenced with elementary techniques like case-based and rule-based approaches, gradually developed to have more sophisticated methodologies, prominently featuring machine learning (ML) techniques. The initial emphasis was primarily on discerning the input-output relationships, with less consideration for understanding the intricate when mentioning the machine learning domain's supervision, processes generating the output. The current investigation focuses on the overarching aim of evaluating the effectiveness of diverse ML techniques, including Support Vector Machine (SVM), XGBoost, RandomForest, Deep Neural Network, and LightGBM. This assessment specially focuses on their ability to discern the malicious attributes embedded in DNS logs, relying on the extraction of specific features. The purpose is to delineate the nuanced efficacy of each ML technique, contributing to the accurate identification and classification of malicious domains within the dynamic landscape of cybersecurity.

The operational process of the Domain Name System (DNS) can be delineated as follows: When a user engages with a website and initiates a search, the DNS system navigates to the nearest root name server to ascertain the answers corresponding to the user's "requested

query." Subsequently, the root server interfaces with the Top-Level Domain (TLD) servers to assemble the constituent parts of the domain name, thereby establishing the complete IP address of the website. Examples of TLDs include, but are not limited to, .com, .org, and .edu. Furthermore, local DNS servers facilitate the oversight of internet traffic by Internet Service Providers (ISPs). This procedural orchestration ensures that users are directed to the correct IP addresses corresponding to their requested queries, contributing to the efficient and accurate functioning of the DNS infrastructure. The detailed workflow of how DNS works in described in Figure 1 below.

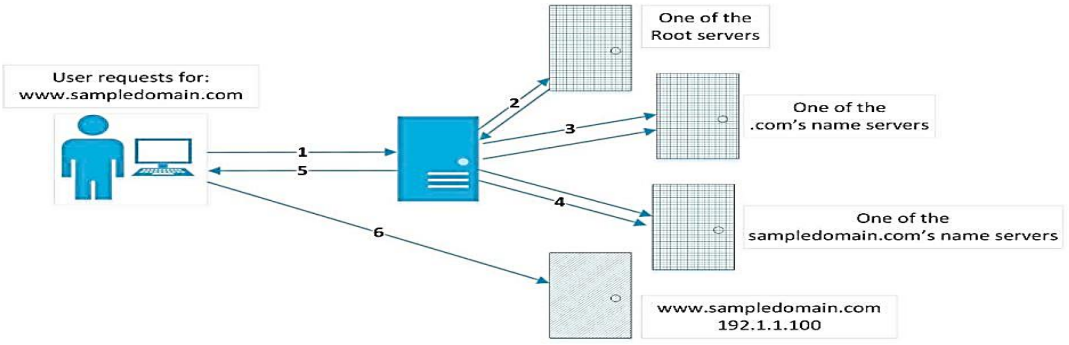

Figure 1. The Workflow of General DNS System

Our study has made noteworthy contributions in the pursuit of advancing cybersecurity. First, a development of a Tailored Machine Learning Model which has made a specialized machine learning model adept at discerning malicious domains within a comprehensive dataset including both malicious and benign domains. Second, the Algorithmic Analysis for AI-based Firewall has conducted a meticulous comparative analysis involving five distinct algorithms, culminating in the conceptualization and development of an advanced artificial intelligence (AI)-based firewall. Furthermore, the structure of the paper is divided into five major parts. To start with, the introduction Provides an overview of the study's objectives and significance. Then, the literature review, section two, offers insights into recent advancements in detecting malicious DNS activities, laying the foundation for the current study. Next, the methodology, section three, delves into the comprehensive methodology, elucidating crucial considerations like dataset selection, data preparation, feature extraction, and a detailed algorithmic description. Moreover, results and evaluation, section four, present the findings derived from the study, accompanied by a detailed exposition of the evaluation metrics employed to assess the model's performance. Finally, the conclusion, section five, summarizes the study's key outcomes, implications, and concluding remarks, contributing to the broader discourse on cybersecurity.

2. Literature Review

In the context of existing research studies, a multitude of studies have explored innovative technologies to discern the malignancy or benign nature of domains. Recent investigations have prominently focused on leveraging deep learning architectures to enhance existing methodologies. Similarly, a substantial body of literature has centered on domain

classification by using the machine learning techniques. This section lists some comprehensive reviews of noteworthy studies employing machine learning methods for domain classification.

Han and Zhang [9] have proposed, in their seminal work, the CLEAN method as an innovative approach for benign domain detection. Utilizing a passive DNS approach, the CLEAN method comprises three integral stages: data pre-processing, stability detection, and the application of a Naïve Bayesian classifier. The data pre-processing phase involves the meticulous filtration of a substantial volume of irrelevant domains. Subsequently, stability detection contributes to the identification of additional domain names, followed by the Naïve Bayesian classifier discerning benign domains within a vast dataset. Experimental validation based on authentic data from a province in China demonstrated the efficacy of the CLEAN model, achieving an impressive average accuracy of 92.2% and an average recall percentage of 82.1%. Noteworthy features employed for classification, drawn from previous studies, encompass specific TTL ranges, TTL variance, length of LMS percentage, average TTL, TTL changes, the number of different TTLs, and numerical characters' percentage. The CLEAN model has showcased significant efficacy despite having a lower recall value attributed to disparities between domain names detected by stability detection in the verification sample and those in the experimentation sample.

In a distinct scholarly endeavor, Antonakasis et al. [10] have introduced the Notos model, conceived as a comprehensive reputation system for DNS. Employing passive DNS data, this model systematically analyzes domain features and constructs a framework incorporating both malicious and benign domains. Subsequently, the framework is utilized to generate a reputation score for a new domain, signifying its potential malicious or benign nature. The experimentation phase involved approximately 1.5 million users engaged in collecting DNS traffic within a large Intrusion Prevention System (IPS) network. In fact, the results have shown the exceptional abilities of the model to accurately identify malicious domains, achieving a minimal 0.38% false positive rate and an outstanding accuracy of 96.8%. The features chosen for this investigation were systematically classified into three distinct categories: network-based features, zone-based features, and evidence-based features. The reputation engine operates in both offline and online training modes.

Furthermore, Khan et al. [11] have introduced a nuanced model for malicious domain detection, utilizing explainable artificial intelligence. Initial exploration involved various machine learning techniques, progressing to ensemble models such as CatBoost, Adaboost, and Extreme Gradient Boosting (XGB). Rigorous testing on a meticulously pre-processed dataset, coupled with Sequential Forward Feature Selection for feature optimization, demonstrated the robust performance of machine learning algorithms in distinguishing between malicious and benign domains. Particularly noteworthy was the Extreme Gradient Boosting model, achieving an outstanding accuracy rate of 98.18%, underscoring the effectiveness of their approach in enhancing precision within the realm of cybersecurity.

Marques, et al. [12] have presented a study which focuses on the development and implementation of a Machine Learning (ML)-based DNS firewall solution aimed at enhancing the detection of malicious domain requests in real-time. Leveraging a dataset with 34 features and 90,000 records derived from authentic DNS logs and enriched through Open-

Source Intelligence (OSINT) sources, the research undergoes exploratory analysis, data preparation, and applies various supervised ML algorithms. The results demonstrate accuracy rates between 89% and 96%, with a classification time ranging from 0.01 to 3.37 seconds. The study contributes to research by providing a publicly available dataset and a replicable methodology for other researchers. In terms of a practical solution, the work lays the foundation for an in-band DNS firewall. Simultaneously, the CART algorithm proves optimal, and the study introduces considerations on the effectiveness of automatic ML processes. The proposed DNS firewall, which does not impact the core DNS service, holds promise for real-world applications, with acknowledgment of the need for further testing in diverse scenarios to enhance robustness.

Mahdavifa, et al. [13] have addressed the pervasive threat of malicious domains and the imperative need for timely detection and classification. The authors highlight the historical challenges of relying on blacklists for domain detection and underscore the evolving role of machine learning techniques in enhancing detection capabilities. The study suggests a robust system based on three distinct categories of features: DNS statistical, lexical, and third-party features extracted from deep analysis of DNS traffic. The authors present a substantial 13,011 malicious samples and dataset comprising 400,000 benign, mirroring real-world scenarios. The methodology involves training and validating a classification model, with k-Nearest Neighbors (k-NN) achieving high performance, particularly an impressive 99.4% F1-score for imbalanced data ratio (97/3%). Feature evaluation using information gain analysis identifies third-party features as pivotal, constituting 58% of the top 13 influential features. The authors release their dataset and propose future work to enrich feature sets, demonstrating the article's comprehensive contribution to advancing the field of malicious domain classification using DNS traffic analysis.

3. Methodology

This paper focuses on the categorization of domains into either malicious or benign categories. The classification methodology involves training on a contemporary DNS dataset through the utilization of supervised machine learning algorithms. The schematic representation in Figure 1 delineates the essential steps the model will execute to categorically classify the provided DNS logs as either malicious or non-malicious. These procedural steps include reading the dataset, preprocessing the dataset, executing feature selection, and employing both machine learning algorithms and deep neural networks.

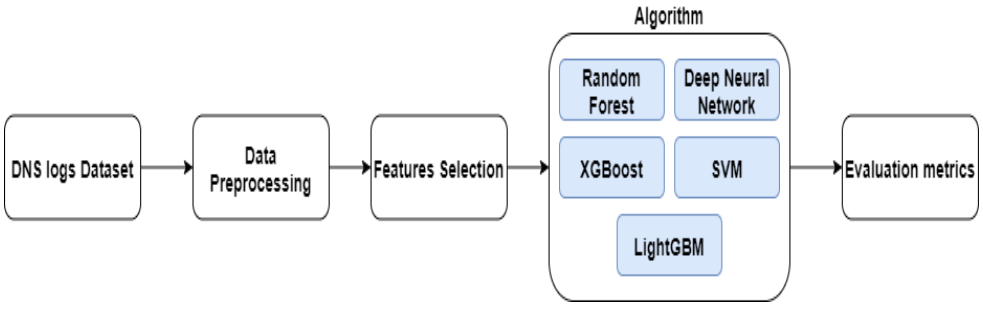

Figure 2. General Workflow of the Proposed Model for DNS logs Classification

3.1 Dataset

The dataset employed for this research has been meticulously curated to facilitate a thorough supervised machine learning analysis aimed at discerning between malicious and non-malicious domain names. Carefully constructed from publicly available DNS logs encompassing both categories, the dataset comprises a total of 90,000 domain names. A deliberate effort has been made to maintain a balanced distribution, allocating 50% of the dataset to non-malicious domain names and the remaining 50% to malicious ones [14].

Within this dataset, a myriad of features has been extracted from the domain names to provide a rich set of attributes for analysis. Within the domain of critical attributes, there exist foundational characteristics integral to the analysis of domain names. These attributes encompass intrinsic elements, notably the domain name itself, entropy, the occurrence of unconventional characters, and the length of the domain name. These fundamental features serve as pivotal indicators in the comprehensive examination of domains within the context of our research. Additionally, data enrichment techniques, such as Open-Source Intelligence (OSINT), have been employed to obtain supplementary features. These features encompass domain creation date, IP address, open ports, geolocation, and registration information.

The comprehensive set of 34 features encompasses a diverse array of aspects, ranging from DNS response details and organizational associations to top-level domains, reputation scores, character ratios, and sequence lengths. This multifaceted feature set is designed to offer nuanced insights into the characteristics of the domains under consideration. Through this meticulous approach, the dataset is poised to contribute to a robust analysis for effectively distinguishing between malicious and non-malicious entities in the realm of DNS.

TABLE I. FEATURES IN THE DATASET FOR MALICIOUS AND NON-MALICIOUS DOMAIN CLASSIFICATION

No.	Feature	Description
1	Domain	Domain name
2	DNSRecordType	DNS record type queried
3	MXDnsResponse	Response from a DNS request for the record type MX
4	TXTDnsResponse	Response from a DNS request for the record type TXT
5	HasSPFInfo	Presence of Sender Policy Framework attribute
6	HasDkimInfo	Presence of Domain Keys Identified Email attribute
7	HasDmarcInfo	Presence of Domain-Based Message Authentication
8	IP	IP address for the domain
9	DomainInAlexaDB	If the domain is registered in the Alexa DB
10	CommonPorts	Presence of the domain on common ports
11	CountryCode	Country code associated with the IP of the domain
12	RegisteredCountryCode	Country code defined in the domain registration process
13	CreationDate	Creation date of the domain
14	LastUpdateDate	Last update date of the domain
15	ASN	Autonomous System Number for the domain
16	HttpResponseCode	HTTP/HTTPS response status code for the domain
17	RegisteredOrg	Organization name associated with the domain
18	SubdomainNumber	Number of subdomains for the domain
19	Entropy	Shannon Entropy of the domain name
20	EntropyOfSubDomains	Mean value of the entropy for the subdomains
21	StrangeCharacters	Number of characters different from [a-zA-Z]
22	TLD	Top Level Domain for the domain

23	IpReputation	Result of the blocklisted search for the IP
24	DomainReputation	Result of the blocklisted search for the domain
25	ConsoantRatio	Ratio of consonant characters in the domain
26	NumericRatio	Ratio of numeric characters in the domain
27	SpecialCharRatio	Ratio of special characters in the domain
28	VowelRatio	Ratio of vowel characters in the domain
29	ConsoantSequence	Maximum number of consecutive consonants in the domain
30	VowelSequence	Maximum number of consecutive vowels in the domain
31	NumericSequence	Maximum number of consecutive numbers in the domain
32	SpecialCharSequence	Maximum number of consecutive special characters in domain
33	DomainLength	Length of the domain
34	Class	Class of the domain (0: malicious, 1: non-malicious)

Table I furnishes a comprehensive overview of the diverse features incorporated within the dataset utilized for the supervised machine learning analysis. The primary objective of this analysis is to classify domain names into distinct categories, specifically distinguishing between malicious and non-malicious entities. Each feature encapsulated in the table plays a crucial role in providing valuable insights into the nuanced characteristics of the domains under consideration.

These features collectively serve as the foundational elements for the development of robust machine learning models tailored for cybersecurity applications. By harnessing the information gleaned from these features, the models can be finely tuned to enhance their efficacy in accurately classifying domain names, thus contributing to the broader objectives of cybersecurity analysis and threat detection.

Figure 3. Class Distribution of the Dataset

Figure 3, titled "Class Distribution of the Dataset," provides a representation of the balanced distribution of classes within the dataset, particularly focusing on threat detection. The box plot clearly depicts an equitable distribution, with an equal representation of malicious and non-malicious domain names, each constituting 50% of the total data.

This balanced class distribution is a crucial aspect in the context of threat detection as it ensures a fair and unbiased representation of both classes. The equal distribution of malicious and non-malicious instances facilitates a robust and evaluation training of machine learning models designed for effective domain classification. By maintaining parity between the two classes, the dataset enables models to learn and generalize patterns without being skewed towards one class over the other. Consequently, the balanced class distribution depicted in Figure 2 lays a solid foundation for the development of accurate and reliable machine learning models for threat detection in the domain classification task.

3.2 Data Preprocessing

In the preprocessing phase, a series of crucial steps were meticulously executed to optimize the dataset for efficient machine learning model training. It included the Column Removal where certain columns deemed non-contributory to the classification task were strategically eliminated. Notable exclusions encompassed 'RegisteredCountry,' 'RegisteredOrg,' 'CountryCode,' 'Domain,' 'DNSRecordType,' and 'TLD.' Also, Handling Null Values existed which follows the removal of specified columns, a thorough check for null values was conducted. Fortunately, the dataset, post-column removal, exhibited the absence of any null values. Then follows the Boolean Column Conversion aiming to streamline the model training process, boolean columns, including 'TXTDnsResponse,' 'MXDnsResponse,' 'HasSPFInfo,' 'HasDmarcInfo,' 'DomainInAlexaDB,' 'CommonPorts,' 'HasDkimInfo,' 'IpReputation,' and 'DomainReputation,' underwent conversion to numerical values (0 for False, 1 for True). After that, Min-Max Normalization lists all numerical features within the dataset underwent min-max normalization. This crucial step ensures uniformity in scale across all features, preventing any singular feature from unduly influencing the learning process. Train-Test Split where the preprocessed dataset was judiciously partitioned into distinct training and testing sets. The training set served as the foundation for training machine learning models, while the testing set was reserved for evaluating model performance. Lastly, the above meticulous sequence of steps collectively contributes to the creation of a pristine, well-structured dataset. Such a dataset, marked by its cleanliness, balance, and structural integrity, proves highly conducive to training robust machine learning models. The ultimate goal is to enable the effective classification of domain names into either malicious or non-malicious categories.

3.3 Feature selection

Feature selection stands as a pivotal phase in the realm of machine learning, dedicated to discerning and preserving the most informative attributes that wield substantial influence over a model's predictive prowess. The employed feature selection algorithm leveraged the chi-squared test methodology via the SelectKBest method. This strategic approach was undertaken to pinpoint the utmost relevant features crucial for accurate prediction of DNS threat classifications. Within the expansive array of features, the algorithm decisively identified 14 elements as paramount in their influence, pivotal for effectively distinguishing between benign and malicious DNS activities which are divided as the follow:

- MXDnsResponse
- TXTDnsResponse

- HasSPFInfo
- CommonPorts
- CreationDate
- LastUpdateDate
- HttpResponseCode
- StrangeCharacters
- IpReputation
- ConsonantRatio
- NumericRatio
- VowelRatio
- NumericSequence
- DomainLength

Collectively, these characteristics furnish pertinent information to machine learning models, facilitating precise predictions. The process of feature selection serves to diminish dimensionality, enhance model interpretability, and potentially augment the model's ability to generalize effectively to previously unseen data. The chosen features encapsulate pivotal facets of DNS-related data, encompassing response characteristics, content types, security-related details, and structural attributes. These elements are considered indispensable for discerning between ordinary and malicious DNS activities.

3.4 Machine Learning Models

a) XGBoost:

The XGBoost algorithm, recommended by Chen and Guestrin [15], is founded on the GBDT structure. In contrast to GBDT, XGBoost incorporates a regularization term in its objective function to prevent overfitting. The main objective function is described a

$$O = \sum_{t-1}^{n} L\left(y_1, F(x_i)\right) + \sum_{k-1}^{t} R(f_k) + C$$

Were $R(f_k)$ shows the regularization term at iteration k, and C being a constant that can be electively.
Regularization term $R(f_k)$ written as,

$$R(f_k) = \alpha H + \frac{1}{2}\eta \sum_{j-1}^{H} w_j^2$$

Where α is the complexity of leaves, H denotes the number of leaves, η signifies penalty variable, and ω_j utput results in each leaf node. Leaves denote the expected categories based on classification criteria, w eaf node denotes the tree node which cannot be divided.

Furthermore, unlike GBDT, XGBoost employs a second-order Taylor series of main functions rathe rst-order derivative. If the loss function is the mean square error (MSE), then the main function may be wri

$$O = \sum_{t-1}^{n} \left[p_t \omega_{q(x_t)} + \frac{1}{2} \left(q_t \omega_{q(x_t)}^2 \right) \right] + \alpha H + \frac{1}{2} \eta \sum_{j-1}^{H} \omega_j^2$$

Where $q(x_i)$ is a function that maps data points to leaves, g_i and h_l represents loss function's first a lerivatives, respectively.

The final loss value is calculated by adding all of the loss values together. Because samples orresponds to nodes of leaf, the ultimate loss value can be calculated by adding loss values of the leaf n esult, the main function can be written as:

$$O = \sum_{j=1}^{T} \left[p_j \omega_j + \frac{1}{2} (Q_j + \eta) \omega_j^2 \right] + \alpha H$$

where $P_f = \sum_{i \in I_j} p_l, Q_j = \sum_{i \in I_j} q_l$, and I_j are the total number of samples in leaf node j.

b) Random Forest

Random Forest (RF) has emerged as a widely adopted machine learning technique renowned for its simplicity and versatility. Proposed by Breiman in 2001, RF serves as a supervised learning approach applicable to both classification and regression tasks [16]. This integrated learning method harnesses the strength of multiple decision trees (DT) to augment prediction accuracy, employing techniques such as majority voting or mean aggregation, depending on the specific task.

Given an input dataset characterized by values Q=q_1, q_2, q_3, ..., q_n, with n denoting the dataset's size, an RF model is constituted by a set of T trees, denoted as T_1(Q), T_2(Q), T_3(Q), ..., T_n(Q). The forecasted results of these decision trees are represented as (R_1)ˆ, (R_2)ˆ, ..., (R_n)ˆ. In the context of regression tasks, the final output of the RF model is determined by averaging the prediction outcomes across all individual trees.

The process of constructing decision trees within Random Forests (RF) involves partitioning initial training sets into smaller subsets. At each split, only a random subset of predictive elements is chosen. To control the growth of trees and prevent indefinite expansion, stopping criteria such as the Gini Diversity Index, Root Mean Square Error (RMSE), or Mean Squared Error (MSE) are incorporated. In the final RF model, trees that exhibit accurate predictions are retained, while those with lower predictive power are excluded. This

systematic methodology guarantees the creation of an efficient ensemble model capable of providing resilient and accurate predictions across a variety of scenarios.

c) Support Vector Machine (SVM)

In 1995, Vapnik introduced Support Vector Machines (SVM), a learning algorithm widely embraced for its efficacy in addressing both linear and nonlinear regression problems [17]. Acknowledged for its capacity to navigate high-dimensional feature spaces, SVM offers robust and dependable predictions, demonstrating notable resistance to noise [18] [19]. The literature extensively documents numerous successful implementations of SVM across various disciplines, adeptly handling challenges in both classification and regression [20] [21] [22].

Succinctly summarizing the foundational theory of SVM involves considering a designated training set {(u_k, v_k), k=1,2,…,n} for an SVM model. In this context, u_k=[u_1k, u_2k, …, u_nk]∈R^(n_k) represents the input data, v_k∈R^(n_m) corresponds to the output data for u_k, and n denotes the number of training samples. The primary goal of SVM is to determine an optimal hyperplane function denoted as f(x), defined by the weight vector w, and the offset b.

d) LightGBM

LightGBM, an acronym for Light Gradient Boosting Machine, represents a robust and highly efficient gradient boosting framework meticulously crafted for distributed and streamlined training processes. Falling within the spectrum of boosting algorithms, renowned for their adeptness in amalgamating weak learners, typically manifested as decision trees, into a formidable learner, LightGBM distinguishes itself through its notable attributes of speed, minimal memory consumption, and elevated efficiency. These characteristics render it particularly well-suited for managing expansive datasets and intricate problem domains.

In contrast to conventional gradient boosting methodologies, LightGBM adopts a leaf-wise growth strategy in lieu of a level-wise approach. This strategic choice results in a diminished number of levels within the trees, thereby optimizing the efficiency and hastening the training process. Furthermore, LightGBM integrates a histogram-based learning technique, enhancing the training phase by discretizing continuous features into bins.

e) Deep Neural Network (DNN)

A Deep Neural Network, often referred to as a neural network or artificial neural network, constitutes a class of machine learning models inspired by the architectural and functional principles of the human brain. DNNs are comprised of layers of interconnected nodes (neurons) designed to process and transform input data, ultimately yielding an output. Noteworthy for their incorporation of multiple hidden layers, these networks possess the capacity to discern intricate patterns and representations from intricate datasets. Deep learning, a subset of machine learning, has garnered significant attention due to its inherent capability to autonomously acquire hierarchical features and representations from unprocessed data.

DNNs exhibit notable proficiency in tasks such as image and speech recognition, natural language processing, and various other intricate pattern recognition challenges. The training

process of a DNN involves iteratively adjusting the weights of the connections between neurons through optimization algorithms and backpropagation. This dynamic adjustment mechanism facilitates the model's adaptability and continual enhancement of its performance over time.

4. Results

The machine learning models, such as Support Vector Machine (SVM), Random Forest, XGBoost, LightGBM, and Deep Learning Neural Network, were subjected to an evaluation aimed at classifying domain names into either malicious or non-malicious categories. The SVM model exhibited a noteworthy accuracy of approximately 96.81%, demonstrating balanced performance across precision, recall, and F1-score for both malicious and benign domains. Surpassing this, the Random Forest model achieved a higher accuracy of about 98.13%, showcasing robust performance across all evaluation metrics. Similarly, the XGBoost model demonstrated an accuracy of approximately 98.21%, accompanied by effective precision, recall, and F1-score for both classes. The LightGBM model attained an accuracy of about 98.12%, maintaining balanced performance metrics. Additionally, the Deep Learning Neural Network yielded a competitive accuracy of 97.85%.

4.1 Support Vector Machine (SVM)

The Support Vector Machine (SVM) model demonstrated a commendable level of performance, achieving an accuracy of approximately 96.81%. Precision, recall, and F1-score metrics for both malicious (Class 0) and benign (Class 1) domains hovered around 97%, indicative of a well-balanced performance. Examination of the confusion matrix presented in Figure 3 reveals the model's adeptness in correctly identifying a significant number of instances for both classes, with only a limited number of misclassifications.

In the graphical representation, correct classifications are represented by a dark blue color, while failed classifications are denoted by a light blue color. For instance, the SVM accurately classified 8567 samples as benign, which were indeed benign, and similarly, correctly identified 8859 samples as malicious. Notably, the model exhibited a low misclassification rate, failing to correctly classify only 287 samples between benign and malicious. This result is particularly noteworthy given the overall success achieved in other aspects of the classification task.

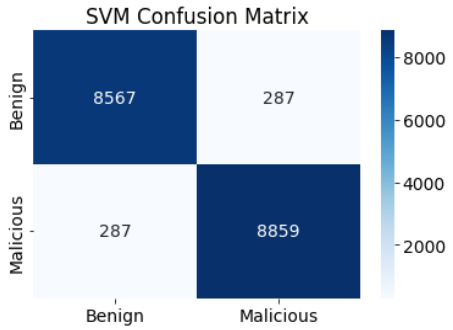

Figure 4. SVM Confusion matrix

4.2 Random Forest

The Random Forest model exhibited superior accuracy at approximately 98.13% compared to the SVM. Precision, recall, and F1-score for both classes also hovered around 98%, indicating robust performance. An examination of the confusion matrix in Figure 4 underscores the model's proficiency in accurately classifying instances for both benign and malicious classes, with a minimal number of misclassifications.

Specifically, the Random Forest correctly classified 8,649 samples as benign, accurately identifying them as such. Similarly, it correctly identified 9,015 samples as malicious, aligning with their true malicious nature. However, the model encountered challenges in classifying 205 samples that were genuinely benign but were misclassified as malicious. Conversely, 131 samples that were truly malicious were erroneously labeled as benign. These results underscore the Random Forest's heightened accuracy compared to the SVM, providing more reliable data.

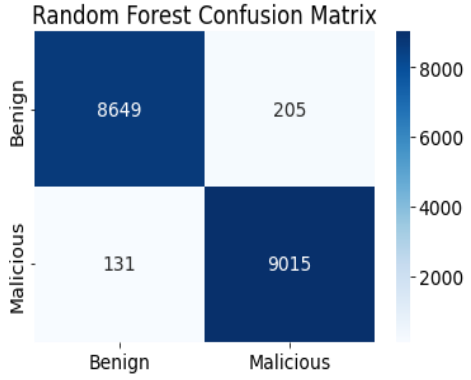

Figure 5. Random Forest Confusion Matrix

4.3 XGBoost

The XGBoost model demonstrated a notable accuracy of approximately 98.21%, with precision, recall, and F1-score consistently hovering around 98% for both classes. The accompanying confusion matrix depicted in Figure 5 underscores its robust performance, accurately categorizing a substantial number of instances for both benign and malicious classes, with minimal misclassifications.

Specifically, the XGBoost model accurately classified 8,665 samples as benign, which were genuinely benign, and correctly identified 9,012 samples as malicious, reflecting their true nature. However, it encountered challenges in classifying 189 instances that were truly benign, misclassifying them as malicious, and erroneously classified 134 truly malicious samples as benign. This analysis substantiates the superiority of the XGBoost model over Random Forest and SVM, affirming its accuracy and reliability in providing trustworthy data.

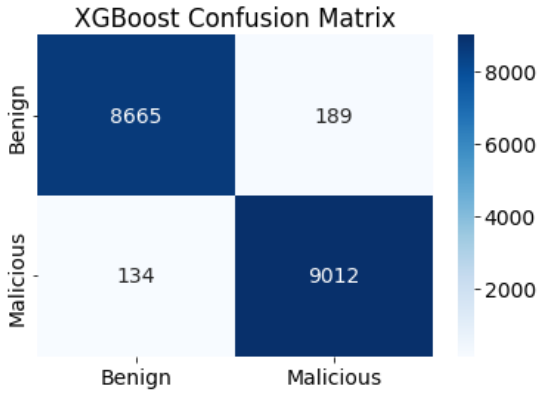

Figure 6. XGBoost Confusion Matrix

4.4 LightGBM

The LightGBM model demonstrated a commendable level of performance, achieving an accuracy rate of approximately 98.12%. Consistent with other models, precision, recall, and F1-score for both classes hovered around 98%. The accompanying confusion matrix, presented in Figure 6, visually conveys the model's effective classification, revealing a minimal number of misclassifications.

Specifically, LightGBM accurately classified 8,640 samples as Benign that were indeed benign, and similarly, correctly identified 9,021 samples as Malicious that were genuinely malicious. However, it exhibited challenges in classifying 214 samples that were truly benign as malicious and misclassified 125 samples that were genuinely malicious as benign. This observation underscores that, while LightGBM falls short of the accuracy achieved by XGBoost and Random Forest, it outperforms Support Vector Machines (SVM) in terms of classification accuracy.

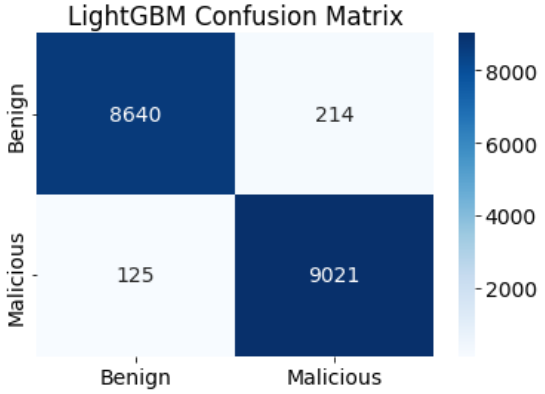

Figure 7. LightGBM Confusion Matrix

4.5 Deep Learning Neural Network

By exhibiting a commendable level of performance of the Deep Learning Neural Network, it appear to be an impressive accuracy rate of 97.85%. The precision, recall, and F1-scores for

both classes approached an exceptional 98%, as meticulously detailed in the comprehensive confusion matrix presented in Figure 7. This matrix underscores the model's efficacy in achieving accurate classification while minimizing instances of misclassification.

Within the dataset, the DNN showcased its robust capabilities by accurately classifying 8,623 samples as Benign, all of which were verifiably benign, and 8,978 samples as Malicious, all of which were unequivocally malicious. Nevertheless, the model revealed certain limitations, manifesting in the misclassification of 231 samples as benign when they were, in fact, benign, and 168 samples as malicious when they were genuinely benign.

This nuanced analysis highlights that, while the DNN demonstrates proficiency in classification tasks, it falls short of the accuracy levels attained by other formidable models such as XGBoost, Random Forest, and LightGBM. Interestingly, the DNN outperforms Support Vector Machine (SVM) in terms of classification accuracy, yet there remains room for improvement to align with the exemplary standards set by other advanced machine learning models.

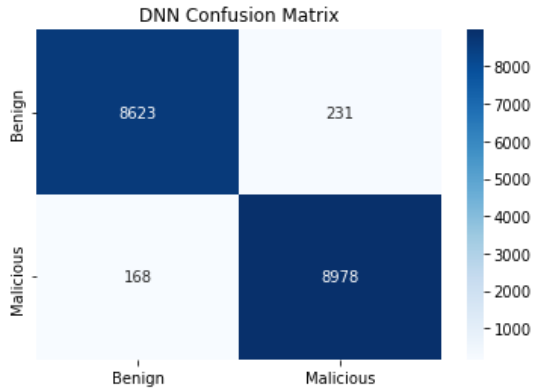

Figure 8. Deep Neural Network Confusion Matrix

Figure 9 bellow depicts the training progression of the Deep Learning (Neural Network) model across 20 epochs, where each epoch corresponds to a complete iteration through the entire training dataset. The graph illustrates the fluctuations in both training and validation loss (measured using binary cross-entropy) as well as accuracy at each epoch. The loss metric quantifies the disparity between predicted and actual values, while accuracy serves as a measure of the model's overall correctness. In the initial epochs, notable enhancements in the model's performance are evident, as reflected by a simultaneous reduction in both training and validation loss. Following this initial phase, the model reaches a point of stability, attaining a high level of accuracy. This visual representation offers valuable insights into the convergence and generalization performance of the model throughout the duration of the training process.

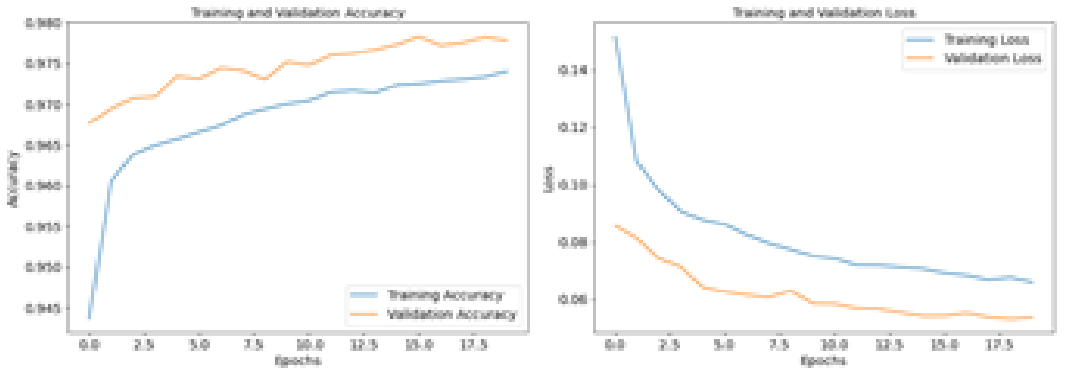

Figure 9. The training progression of the Deep Learning (Neural Network) model

The Area Under the Curve (AUC) serves as a quantitative metric for assessing the efficacy of binary classification models, commonly applied within the context of receiver operating characteristic (ROC) curves. The ROC curve provides a graphical representation of the compromise between the true positive rate (sensitivity) and false positive rate (1-specificity) across different classification thresholds. AUC quantifies the area beneath this curve, offering a singular value for evaluating a model's capacity to differentiate between the two classes.

The Support Vector Machine (SVM) model demonstrates commendable discriminatory capability, attaining an AUC of 0.9681, indicative of its proficiency in discerning between malicious and non-malicious instances. The model exhibits satisfactory performance in the accurate classification of positive and negative cases.

XGBoost showcases outstanding discriminatory power, registering an AUC of 0.9820. Its notable ability to distinguish between the two classes underscores its effectiveness as a classification model, contributing to its high performance.

The Random Forest model achieves a notably elevated AUC of 0.9813, signaling robust performance in discriminating between instances of malignancy and non-malignancy. The ensemble nature of the model significantly contributes to its overall effectiveness.

The Deep Neural Network (DNN) model displays a robust AUC of 0.9778, indicative of its capability to make precise predictions and distinguish between the two classes. While marginally lower than some other models, it remains a formidable performer.

LightGBM attains an AUC of 0.9811, underscoring its effectiveness in classification tasks. The model excels in discerning between malicious and non-malicious instances, further substantiating its commendable overall performance.

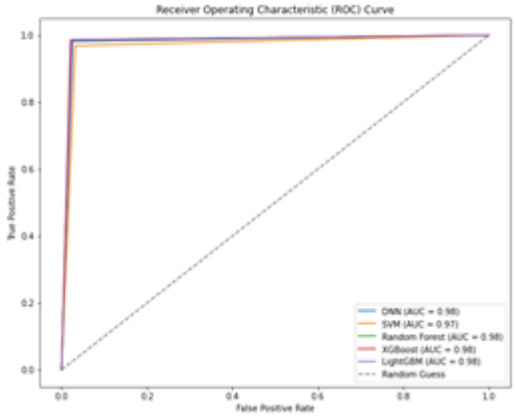

Figure 10. Receiver Operating Characteristics (ROC)

This graphical representation illustrates the Receiver Operating Characteristic (ROC) curves for each model, delineating the true positive rate against the false positive rate across diverse classification thresholds. The Area Under the Curve (AUC) values associated with each curve serve as quantitative measures of the discriminative performance of the respective models. A larger AUC is indicative of a superior model with enhanced discriminatory capabilities. In the context of this figure, the visual depiction underscores that XGBoost attains the highest AUC, underscoring its exceptional discriminatory power. Following closely are Random Forest and LightGBM, both exhibiting commendable AUC values, suggesting robust performance. Meanwhile, Support Vector Machine (SVM) and Deep Neural Network (DNN) also demonstrate proficient performance, albeit with slightly lower AUC values, affirming their effectiveness in classification tasks. This nuanced analysis of the ROC curves and associated AUC values provides a comprehensive perspective on the relative strengths of each model in terms of their discriminative abilities.

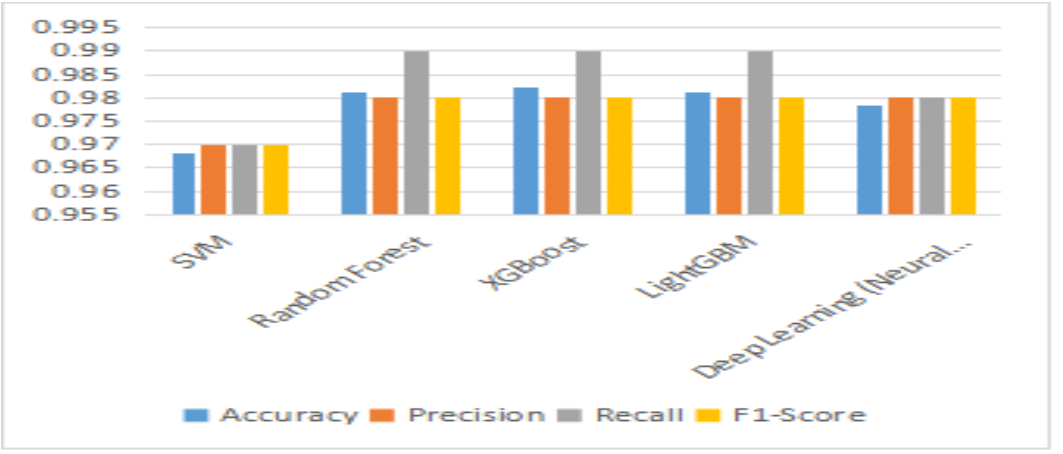

Figure 11. Bar Plot Representation of the Performance Metrics of Our Algorithms

The presented bar plot illustrates the outcomes as follows: Correct classifications are represented by a dark blue color, while failed classifications are indicated by a light blue

color. For instance, the SVM accurately classified 8,567 samples as Benign, correctly identifying them as truly benign. Similarly, it successfully classified 8,859 samples as Malicious, accurately identifying them as truly malicious. Notably, the model exhibited a low failure rate in classifying only 287 samples between benign and malicious, indicating a high level of accuracy compared to other instances of successful classifications.

4.6 Results Discussion

Upon a comprehensive comparative analysis of our study's outcomes against those of other pertinent investigations, a discernible trend emerges, unequivocally affirming the superior efficacy of the algorithms employed in our research. Noteworthy is the exemplary performance of our Support Vector Machine (SVM), attaining an accuracy of 0.9681, a marked improvement over the 0.910 reported in the DNS Firewall Based on Machine Learning study [12] and the 61.2% recorded in the Classifying Malicious Domains using DNS Traffic Analysis study [13].

Moreover, our study establishes a conspicuous advantage in terms of precision, recall, and F1-score, pivotal indicators of accuracy, when juxtaposed with the aforementioned referenced studies. This notable superiority extends uniformly across all algorithms examined in our research, namely Random Forest, XGBoost, LightGBM, and Deep Learning, attesting to the consistent and elevated performance of our models across various metrics in contrast to those presented in the cited papers.

Nevertheless, a judicious interpretation of these results is imperative, considering the potential contingencies affecting actual performance. Factors such as dataset selection, feature extraction methodology, and model configuration settings wield considerable influence. Furthermore, it is pertinent to acknowledge that the Classifying Malicious Domains using DNS Traffic Analysis study [13] utilized a distinct dataset, necessitating a nuanced consideration of observed disparities and caution in drawing direct comparisons.

In conclusion, our findings substantiate a substantial enhancement in the classification of DNS threats through the adept application of machine learning techniques. As we celebrate the advancements achieved, it is incumbent upon us to remain circumspect and recognize the nuanced intricacies that may influence performance, thereby fostering a more nuanced understanding of the evolving landscape of cybersecurity threats and the corresponding efficacy of our proposed solutions.

Table 2. Summary Of Results

Algorithm	Our Paper Accuracy	Paper [12]	Paper [13]
SVM	0.9681	0.910	61.2
LR	-	0.913	76.9
LDA	-	0.907	-
KNN	-	0.957	94.8
CART	-	0.961	-
NB	-	0.897	-
Random Forest	0.9813	-	-
XGBoost	0.9821	-	-
LightGBM	0.9812	-	-
Deep Learning (Neural Network)	0.9785	-	-
MLP	-	-	72.2
GNB	-	-	78.2

5. Conclusion

Our research represents a significant advancement in the domain of cybersecurity, particularly addressing the significant role of DNS firewall solutions. The escalating complexity of cyber threats necessitates innovative approaches, and our study has diligently explored the integration of cutting-edge Machine Learning (ML) techniques to enhance the real-time detection of malicious domain requests. The meticulous construction of a dataset, incorporating 34 intricate features and comprising 90,000 meticulously recorded instances derived from authentic DNS logs, enriched with Open-Source Intelligence (OSINT) sources, underscores the rigor and depth of our research methodology. By leveraging this comprehensive dataset, our primary objective was to empower a DNS firewall solution capable of accurately and promptly classifying domain requests as either malicious or benign. Notably, our Support Vector Machine (SVM) achieved an accuracy of 0.9681, surpassing previous studies in the field. This trend persisted across all algorithms investigated, including Random Forest, XGBoost, LightGBM, and Deep Learning, demonstrating consistently elevated accuracy and key performance metrics such as precision, recall, and F1-score. Based on these considerations, our study relates substantively to the discourse on DNS threat classification, offering a robust framework for enhancing cybersecurity protocols. While recognizing the advancements achieved, we focused for ongoing scrutiny and refinement, acknowledging the dynamic nature of the cybersecurity landscape. Lastly, our results underscore the efficacy of integrating machine learning techniques in fortifying DNS firewall solutions, presenting a promising avenue for continued research and implementation in the evolving realm of cybersecurity.

References

1. A. Almusawi and H. Amintoosi, 'Dns tunneling detection method based on multilabel support vector machine', Secur. Commun. Networks, vol. 2018, 2018.
2. Li, K.; Yu, X.;Wang, J. A Review: How to Detect Malicious Domains. In International Conference on Artificial Intelligence and Security; Springer: Cham, Switzerland, 2021; pp. 152–162.
3. DNSWas Not Designed for Security. Available online: https://www.cloudflare.com/learning/insights-dns-landscape/ (accessed on 5 April 2022).
4. Adiwal, Sanjay & Rajendran, Balaji & Shetty, Pushparaj. 2018. Domain Name System (DNS) Security: Attacks Identification and Protection Methods. In SAM'18: International Conference on Security and Management, Las Vegas, USA.
5. Tejaswini Yadav C.Y, Balaji Rajendran, and Rajani P. 2014. An Approach for Determining the Health of the DNS. International Journal of Computer Science and Mobile Computing, Vol.3 Issue.9, September- 2014, pg. 442-449".
6. P., Gopinath & S., Sangeetha & Rajendran, Balaji & Adiwal, Sanjay & Goyal, Shubham & Bindhumadhava, Bapu. (2020). Malicious Domain Detection Using Machine Learning On Domain Name Features, Host-Based Features and Web-Based Features. Procedia Computer Science. 171. 654-661. 10.1016/j.procs.2020.04.071.
7. Yury Zhauniarovich, Issa Khalil, Ting Yu, and Marc Dacier. 2018. A Survey on Malicious Domains Detection through DNS Data Analysis. ACM Computing Survev. 1, 1, Article 1 (May 2018), 35 pages.
8. Kim, T.H.; Reeves, D. A survey of domain name system vulnerabilities and attacks. J. Surveill.

Secur. Saf. 2020, 1, 34–60.
9. C. Han and Y. Zhang, "CLEAN : An approach for detecting benign domain names based on passive DNS traffic," 2017 6th International Conference on Computer Science and Network Technology (ICCSNT), 2017, pp. 343-346, doi: 10.1109/ICCSNT.2017.8343715.
10. Manos Antonakakis, Roberto Perdisci, David Dagon, Wenke Lee, and Nick Feamster. 2010. Building a dynamic reputation system for DNS. In Proceedings of the 19th USENIX conference on Security (USENIX Security'10). USENIX Association, USA, 18.
11. Khan, Irfan & Mirza, Samiha & Alowayed, Alanoud & Anis, Fatima & Aljuaid, Reef & Baageel, Reham & Aslam, Nida. (2022). Interpretable Machine Learning Models for Malicious Domains Detection Using Explainable Artificial Intelligence (XAI). Sustainability. 14. 10.3390/su14127375.
12. C. Marques, S. Malta, and J. Magalhães, "DNS Firewall Based on Machine Learning
13. S. Mahdavifar, N. Maleki, A. Habibi Lashkari, M. Broda, and A. H. Razavi, "Classifying Malicious Domains using DNS Traffic Analysis
14. Marques, Claudio (2021), "Benign and malicious domains based on DNS logs", Mendeley Data, V5, doi: 10.17632/623sshkdrz.5
15. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
16. Breiman, L. Random Forests. Mach. Learn. 2001, 45, 5–32.
17. Cortes, C.; Vapnik, V. Support-vector networks. Mach. Learn. 1995, 20, 273–297.
18. Wadho, S. A., Yichiet, A., Lee, G. M., Kang, L. C., Akbar, R., & Kumar, R. (2023, October). Impact of Cyber Insurances on Ransomware. In 2023 IEEE 8th International Conference on Engineering Technologies and Applied Sciences (ICETAS) (pp. 1-6). IEEE.
19. Hsu, C.W.; Chang, C.C.; Lin, C.J. A Practical Guide to Support Vector Classification; National Taiwan University: Taipei, Taiwan, 2003; Available online: http//www.csie.ntu.edu.tw/~cjlin (accessed on 1 July 2021).
20. Fowler, B. A sociological analysis of the satanic verses affair. Theory Cult. Soc. 2000, 17, 39–61.
21. Barakat, N.; Bradley, A.P. Rule extraction from support vector machines: A review. Neurocomputing 2010, 74, 178–190
22. Wadho, S. A., Yichiet, A., Gan, M. L., Lee, C. K., Ali, S., & Akbar, R. (2024, January). Ransomware Detection Techniques Using Machine Learning Methods. In 2024 IEEE 1st Karachi Section Humanitarian Technology Conference (KHI-HTC) (pp. 1-6). IEEE.
23. Martens, D.; Huysmans, J.; Setiono, R.; Vanthienen, J.; Baesens, B. Rule extraction from support vector machines: An overview of issues and application in credit scoring. Rule Extr. Support Vector Mach. 2008, 80, 33–63.
24. Wadho, S. A., Meghji, A. F., Yichiet, A., Kumar, R., & Shaikh, F. B. (2023). Encryption Techniques and Algorithms to Combat Cybersecurity Attacks: A Review. VAWKUM Transactions on Computer Sciences, 11(1), 295-305.
25. Uslan, V.; Seker, H. Support Vector-Based Takagi-Sugeno Fuzzy System for the Prediction of Binding Affinity of Peptides. In Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Osaka, Japan, 3–7 July 2013; Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2013; pp. 4062–4065