University of Ibadan
Journal of Science and Logics in ICT Research

A Multiclass Model for Adversary Domain Name Classification using Tree-Based AI Classifiers

[1]✉Odigie B. B., and [2]Bernard O. P.
[1]*Department of Computer Science, Benison Idahosa University, Benin City.*
[2]*Department of Computer Science, Federal Polytechnic, Auchi.*

Emails of authors: bodigie@biu.edu.ng, Bernard.femi@auchipoly.edu.ng

Abstract

The rising prevalence of AI-generated adversary (malicious) domain names has escalated the challenge of combating cybercrime, particularly as spamming, phishing, and malware activities become increasingly common online. Traditional approaches, such as blacklisting, binary detection systems, and basic lexical analysis of domain names, prove insufficient for real-time identification of malicious domains across various cyber threat landscapes. This study presents a comprehensive strategy for the multiclass detection of malicious domain names (MDNs) utilizing data mining techniques. It investigates feature engineering processes, including dimensionality reduction and variance inflation factor analysis, to identify and select domain name features that enhance the performance of advanced AI and machine learning classifiers in classifying MDNs. We employed a train/test split ratio and cross-validation methods on the CIC-Bell-DNS2021 public dataset for training some cutting-edge AL/ML classifiers. The findings reveal that tree-based machine learning algorithms, particularly the Extreme Gradient Boosting (XGBoost) algorithm, achieved outstanding results, with a mean accuracy score of 0.9998 (100%). Additionally, regarding execution time, XGBoost displayed a notable advantage, requiring less time to build models, which could significantly influence real-time detection capabilities when implemented as a cybersecurity tool for detecting malicious domain names.

Keywords: *Malicious attacks, Domain Name, Multiclass, Classifiers, Artificial Intelligence*

1.0 Introduction

There has been a notable increase in malicious domain trends since 2017 [1]. This uptick can be attributed to several factors, including the rise of AI-generated domains and malicious activities linked to ChatGPT [2, 3]. Additionally, the growth of malicious subdomains [4], phishing attacks targeting online document and storage platforms [5], and the increasing prevalence of malicious documents across various digital and web applications [6] have all played significant roles in this rising trend.

Furthermore, the use of injected JavaScript malware—including downloaders, web skimmers, crypto-miners, redirectors, and web scams - has also surged [7, 8, 9]. All these nefarious practices have widened the gap for fighting malicious domain name attacks.

Odigie B. B., and Bernard O. P. (2025). A Multiclass Model for Adversary Domain Name Classification using Tree-Based AI Classifiers, *University of Ibadan Journal of Science and Logics in ICT Research (UIJSLICTR)*, Vol. 13 No. 1, pp. 95 - 108

Malicious actors are using AI tools to create sophisticated domains for spamming, malware, and phishing attacks. Traditional detection methods, such as binary classification systems, blacklisting, and traffic analysis, are becoming inadequate for real-time detection of these malicious domains in an evolving threat landscape [11]. It is essential to develop an improved and more effective multiclass Malicious Domain Name (MDN) detection system. Key detection processes include a layered security approach [5], protective DNS [11], advanced URL filtering [10], and continuous updates on threat intelligence related to DNS security [12].

Threat intelligence actors and cybersecurity experts are using advanced AI/ML algorithms and real-time feedback to create detection systems aimed at reducing MDN proliferation [13, 14, 9]. Feature engineering is essential for identifying key domain name features in MDN detection [15, 16, 17]. Several studies have utilized the CIC-Bell-DNS2021 dataset to develop an adversarial domain name detection system that can effectively identify both benign and malicious domains [18, 19, 20, 21]. A comprehensive one-for-all or multi-class detection system is essential to address

threats from malicious domains related to spamware, malware, and phishing in the evolving threat landscape.

This study presents a multi-class Malicious Domain Name (MDN) detection system that leverages advanced AI and machine learning algorithms. It aims to develop a data mining approach to identify features pertinent to MDN classification and deploy the most effective machine learning model for predicting new cases. Several research questions are posed to guide the objectives.

a. What key features enhance the detection of malicious domain names across various categories?
b. How effective are machine learning algorithms in identifying malicious domain names used for spamming, distributing malware, and conducting phishing attacks?
c. Which machine learning algorithms can deliver outstanding performance for a multi-class MDN classification system?

2.0 Related Works

The Domain Name System (DNS) is a crucial component of the Internet's architecture that converts human-readable domain names into machine-readable IP addresses. Consequently, cybercriminals exploit vulnerabilities in the DNS channel to execute malicious activities [16].

In recent times, attackers have employed dynamic DNS techniques such as Fast-Flux and Domain-Flux to obscure the locations of their malicious services, making malicious domain names difficult to detect [22]. Consequently, advanced methods have been developed to detect these malicious domain names, as illustrated in Figure 1. Machine learning-based techniques for analyzing domain names encompass several key approaches:

- Supervised Machine Learning (SML) utilizes labeled datasets for training models [22, 23].
- Unsupervised Machine Learning (UML) applies clustering and anomaly detection to uncover patterns in domain name features [24, 25, 26].
- Deep Learning (DL) employs Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) for a more in-depth analysis of domain features [27, 28, 29, 30]
- Natural Language Processing (NLP) utilizes domain name features for the effective detection of MDNs ([31], [32])

Feature extraction approaches used to analyze domain names (DNs) consist of:

- Lexical Features: Assessing characteristics such as DN length, entropy, and keyword presence [33].
- Syntactic Features: Investigating the structure of DNs, including hyphens, numbers, and special characters [34].
- Semantic Features: Evaluating the meaning of DNs and identifying potentially suspicious keywords [35].

Behavioral Analysis-Based Approaches consist of the following methods:

- DNS Traffic Analysis: This technique aims to detect suspicious patterns, such as rapid changes in domain names ([36], [22]).
- Web Traffic Analysis: This method focuses on identifying malicious domain names, particularly those associated with phishing sites [37].
- Network Traffic Analysis: This approach involves monitoring network traffic to identify communications with known malicious domains[38]

Collectively, these strategies enhance security by effectively identifying potential threats within network communications. The Reputation-Based approach encompasses several strategies. Blacklisting involves blocking known malicious domains [39], while whitelisting reduces false positives by relying on lists of trusted domains [40]. Additionally, reputation scoring assigns scores to domain names based on their historical behavior [41].

The graph-structured approaches, including heterogeneous methods, utilize graph theory to analyze domain names for identifying unknown MDNs [42, 43]. The increase in AI-generated domains used for spamming, phishing, and malware highlights the need for robust security measures to collect and analyze large datasets of malicious domain names (MDNs).

Studies have revealed the superiority of AI/ML approaches for detecting MDNs when sufficient data is presented to these AI models. Cybersecurity actors must leverage these tools to implement cutting-edge tools for MDN detection

[44]. The first public dataset for multi-classification of the malicious domain names is the CIC Bell DNS 2021 dataset, which contains up-to-date real-time DNS-related data that can be useful for flagging a particular DNS request as benign, spam, phishing, and malware [17].

Some recent studies have employed the CIC Bell DNS 2021 to train cutting-edge AI/ML like KNN with a 98.9% accuracy score [17], ImmuneNet with a 99.2% accuracy [45], one-dimension convolutional neural network (1DCNN) with 95.6% accuracy [46], Random forest with 89.9% accuracy [47], BiLSTM with a 92.38% accuracy score [48] and Deep convoluted bi-LSTM network (AConBN) with 99.51% accuracy [44]

3.0 Research Methodology

The study uses a data mining methodology called KDD (Knowledge Discovery in Databases) [49]. Figure 2 presents the process flow for MDNs in this study.

Problem Formulation: The traditional method of DNS filtering that relies on a blacklist of malicious domain names (MDNs) has proven to be inefficient in detecting newly generated malicious domains created by AI. The business objective is to identify key domain features in the evolving threat landscape using advanced AI and machine learning (ML) algorithms. By utilizing machine learning models, we aim to establish a robust feature engineering mechanism that plays a crucial role in enhancing applications within the cybersecurity space. Therefore, understanding the features and data contained in transmitted packets or DNS traffic is essential for identifying potential DNS attacks.

Data Collection: Mahdavifar et al. [17] presented a study on extracting effective and practical features from DNS traffic. They identified three classes of features. The first class includes statistical features, which are extracted from the structure of DNS messages (specifically the answer section) within a designated packet window captured in DNS PCAP files.

The second class consists of lexical features, which are derived from the statistical features and aid in identifying malicious domain names. These domain names are often used by attackers who employ various typosquatting and obfuscation techniques to mimic legitimate domains. The third class comprises third-party features, which encompass the biographical properties of a domain. An overview of these three classes of features is presented in Table 1.

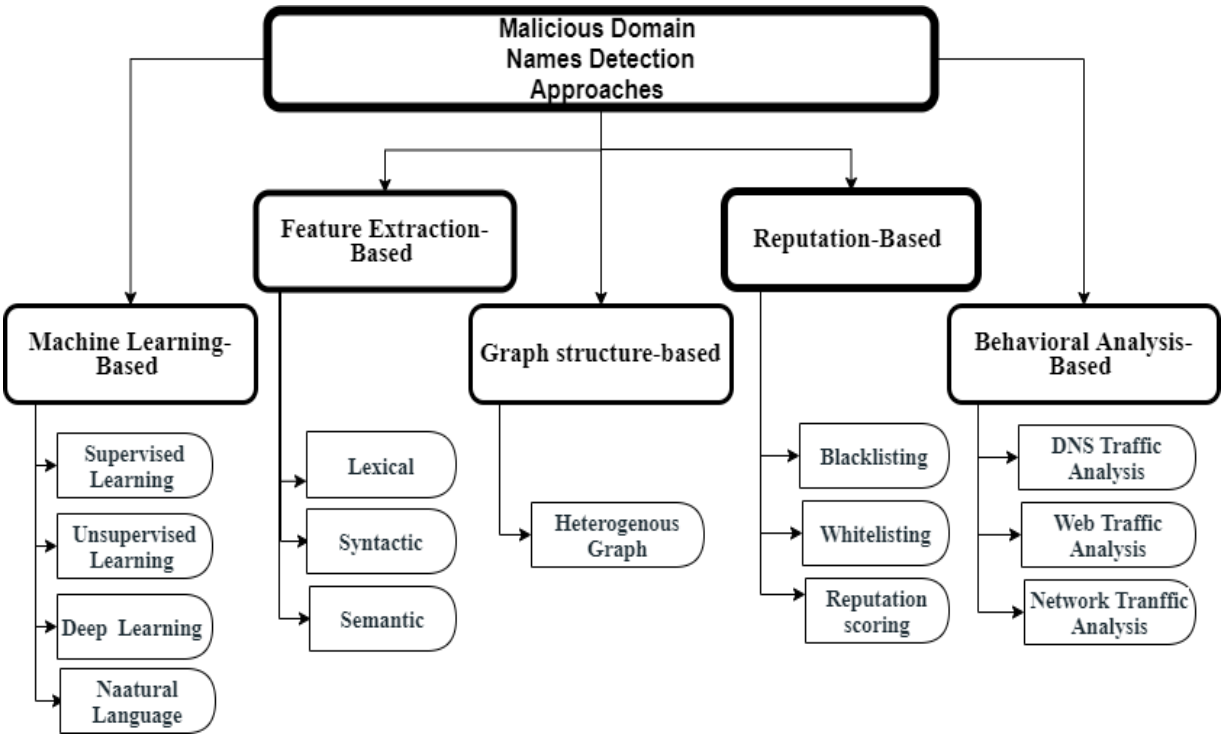

Figure 1: Categorization of Malicious Domain Names Detection Approaches

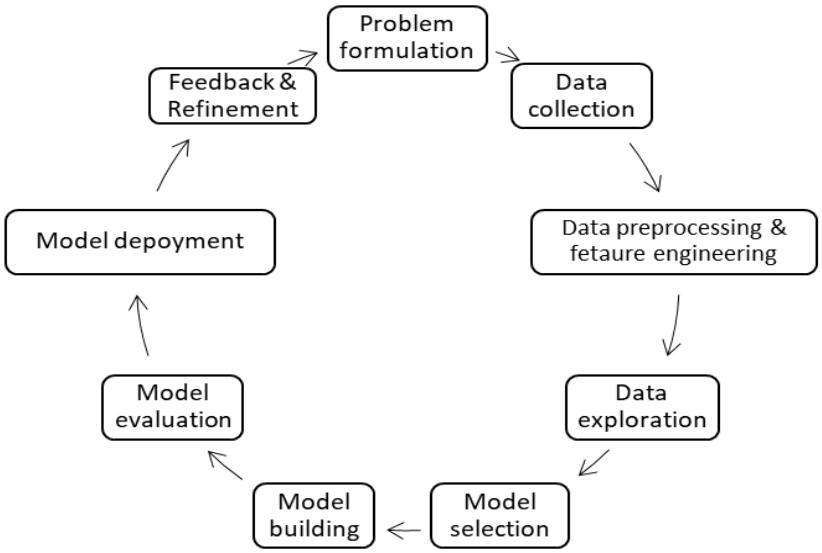

Figure 2: Data Mining Approach for Malicious DNS Classification

Table 1: DNS Statistical Features

SN	Feature name	Description	Type
		DNS statistical features	
1	Unique country	number of distinct country names in the window	Numeric
2	Unique ASN	number of distinct ASN values in the window	Numeric
3	Unique TTL	number of distinct TTL values in the window	Numeric
4	Unique IP	number of distinct IP values in the window	Numeric
5	Unique domain	number of distinct domain values in the window	Numeric
6	TTL means	average TTL in the window	Numeric
7	TTL variance	variance of TTL in the window	Numeric
		Lexical features	
8	Subdomain	Has sub-domain or not	Boolean
9	TLD	Top-level domain	Text
10	SLD	Second-level domain	Text
11	Len	Length of domain and subdomain	Numeric
12	Numeric percentage	Counts the number of digits in the domain and subdomain	Numeric
13	Character distribution	Counts the number of each letter in the domain	Numeric
14	Entropy	Entropy of letter distribution	Numeric
15	1-gram	1-gram of the domain in letter level	Text
16	2-gram	2-gram of the domain in letter level	Text
17	3-gram	3-gram of the domain in letter level	Text
18	Longest word	Longest meaningful word in SLD	Text
19	Distance from bad words	Computes average distance from bad words	Text
20	Typos	Typosquatting	Text
21	Obfuscation	Max value for URL obfuscation	Numeric
		Third-party Features	
22	Domain name	Name of the domain	Text
23	Registrar	Registrar of the domain	Text
24	Registrant name	name of the domain has been registered	Text
25	Creation date time	date and time the domain was created	Date
26	Emails	emails associated with a domain	Text
27	Domain age	Age of a domain	Text
28	Organization	What organization is it linked to	Text
29	State	state the main branch is	Text
30	Country	country where the main branch is	Text
31	Name server count	total number of name servers linked to the domain	Text
32	Alexa rank	Alexa rank of domain	Numeric

Sources of Data: This study uses the CIC-Bell-DNS2021 dataset, which is obtained from the Canadian Institute for Cybersecurity. The dataset consists of different types of domains categorized into four groups: malware, spam, phishing, and benign. Domain data were collected from May 2019 to June 2019, with additional updates made in December 2020 to enhance validation. Table 2 presents statistics on the distribution of these domain classes within the dataset.

Table 2 summarizes the domain name instances, revealing an imbalance between classes: Benign is the most prevalent, while Spam is the least. This imbalance can distort predictive model performance by misleading accuracy metrics, leading to overfitting of the majority class and underfitting of the minority class. It also causes model instability and challenges in selection and generalization. The Synthetic Minority Over-sampling Technique (SMOTE) helps mitigate this class imbalance issue.

Data Preprocessing & Engineering: Categorical features in various class domains are transformed into continuous values using label encoding. Missing values are replaced with the mean of their respective fields, and unnamed columns are removed. New features, such as "average_bad_words" derived from "bad_words" and "average_n_gram" calculated from the means of 1_gram, 2_gram, and 3_gram, are created. A "status" feature differentiates the classes in the dataset. Ultimately, the classes are merged column-wise into a single dataset containing 34 features.

Data Exploration: The merged dataset was analyzed to identify relevant features using correlation analysis. Principal Component Analysis (PCA) reduced the dataset's dimensionality, while the Variance Inflation Factor (VIF) detected multicollinearity among features. Features with a VIF below 5 were retained, enhancing model stability and predictive accuracy.

Model Selection: Machine learning algorithms were trained and validated using a balanced dataset for classifying malicious DNS. Table 3 presents the machine learning models applied.

Model Building and Tuning: This phase involves model building, which includes training, testing, and parameter tuning. After feature engineering, the dataset was divided into training and testing sets using an 80/20 split ratio. Additionally, 5-fold and 10-fold cross-validation techniques were employed to ensure that the models (table 3) generalize well.

Model Evaluation: The performance of different models was assessed using classification and regression metrics shown in Table 4, along with the execution time required for training the models**.**

Where n denotes the total number of samples being tested, TP is true positive, TN is true negative, FP is false positive and FN is false negative, $\bar{y}$is the mean of the target vector, $\hat{y}_i$ is the predicted value of y for observation I and y_i is the actual y value for observation i.

Model Deployment & Operationalization: The best-performing model (BPM) is selected and deployed for operationalization based on the highest accuracy score, lowest RMSE value, and fastest execution time during training.

Table 2: Statistics of the Dataset

Class of domain	Description	Domain instances	% distribution	# of features	Size of file	Reference
Malware	Driven by download, DGA-based botnets, DDos and spyware	8871	17.7	32	5.7	Mahdavifar et al., [17], Savenko et al., [50]
Spam	e-mails	4337	8.6	32	2.8	
Phishing	Malicious links on the website	12702	25.3	32	7.7	
Benign	Web domains	24249	48.3	32	13.3	

.

Table 3: Machine Learning Models Used

Machine learning classifier	Type
AdaBoostClassifier	Ensemble
GradientBoostingClassifier	Ensemble
RandomForestClassifier	Ensemble (decision tree, bagging)
XGBClassifier	Ensemble (decision tree, boosting and Regularization)
DecisionTreeClassifier	Supervised
LogisticRegression	Supervised
KNeighborsClassifier	Supervised
LinearSVC(Support Vector Machine)	Supervised
Naïve Bayes	Supervised
MLPClassifier	Supervised

Table 4: Model Evaluation and Performance Metrics.

Type of Evaluation	Metrics	Formula
Classification	Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
	Recall (sensitivity)	$\frac{TP}{TP+FN}$
	Precision	$\frac{TP}{TP+FP}$
	F-1 Score	$2 \times \frac{Precision \times Recall}{Precision + Recall}$
Regression	Mean absolute Error(MAE)	$\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}$
	Root Mean Square Error (RMSE)	$\sqrt{MSE} = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}}$
	Coefficient of determination (R^2)	$\frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})}{\sum_{i=1}^{n}(y_i - \bar{y})}$

Feedback and Refinement: The deployed system is continually refined based on client feedback, which is sent to security experts for action. Updates to the DNS detection system are shared with users to align with business objectives. The data mining strategy and key components for the proposed multi-class MDN detection system are shown in Figure 3.

4. Implementation and Discussion of Results

4.1 Implementation: The multiclass detection system for malicious domain names was successfully implemented using an Intel Core i7 processor, equipped with 16 GB of RAM and a 300 GB hard disk. Python was the programming language of choice, allowing for efficient development. We leveraged a range of powerful

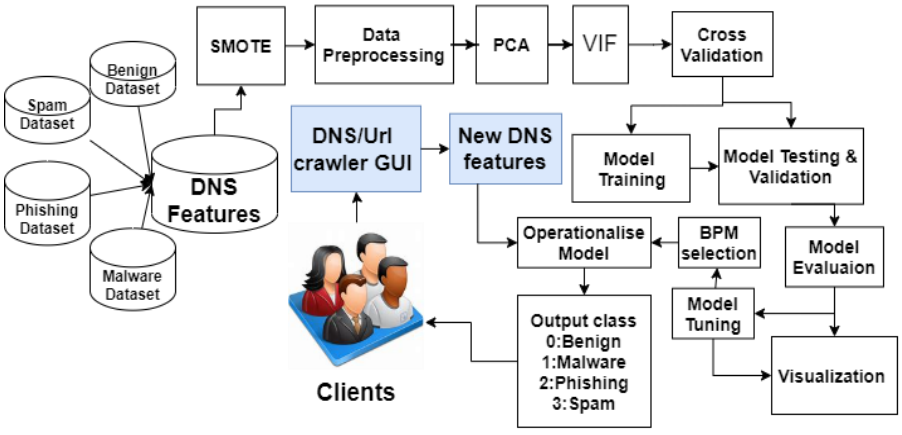

Figure 3: Architecture of the Proposed Multiclass MDN Classification System

Python libraries tailored for specific tasks, including Scikit-Learn for model building, Plotly and Matplotlib for visualization, Pandas for data manipulation, and NumPy for numerical computing. The implementation consisted of two comprehensive phases: Phase 1 encompassed Exploratory Data Analysis (EDA), while Phase 2 concentrated on Model Development, Evaluation, and Deployment.

Phase 1: Exploratory Data Analysis (EPA)

A. Handling Imbalance classes: The class distribution in Figure 4 reveals an imbalance among the sources in the dataset. The balanced dataset, which includes 4,337 instances for each class, is presented in Figure 5 and was utilized.

B. Feature Relationship using Correlation Analysis.

Correlation analysis is essential for understanding the relationships among features in a dataset. Features that are highly correlated with the predictor are considered relevant for model building. Additionally, an analysis of inter-feature correlation helps identify highly correlated independent features that may need to be removed to avoid multicollinearity. Multicollinearity can lead to issues such as model unreliability, imprecision, and ambiguity.

The correlation matrix for the dataset, which consists of 34 features, is presented in Figure 6. Features with an inter-item correlation above 0.8 (greater than 80%) were removed to mitigate their combined impact on the model. For instance, "obfuscate_at_sign" shows a high correlation of 0.86 with "dec_32," leading to its exclusion. Other features have correlation levels below or equal to 0.8, allowing for dimensionality reduction through feature extraction.

C. Feature Dimensionality Reduction using Principle Component Analysis (PCA)

The PCA diagram in Figure 7 illustrates that 33 features extracted from data preprocessing significantly contribute to the variance in the dataset. Therefore, their variance was further analyzed using the Variance Inflation Factor.

D. Feature Selection using Variance Inflation Factor (VIF)

Identifying features with low variance is crucial for building high-precision classification models. Figure 8 displays the features with a VIF lower than five (5). A total of 21 features were retained for model development and evaluation. The Alexa_rank feature exhibited the highest tolerance and the lowest VIF. The features selected based on VIF that are relevant to high-performance models are listed in Table 5.

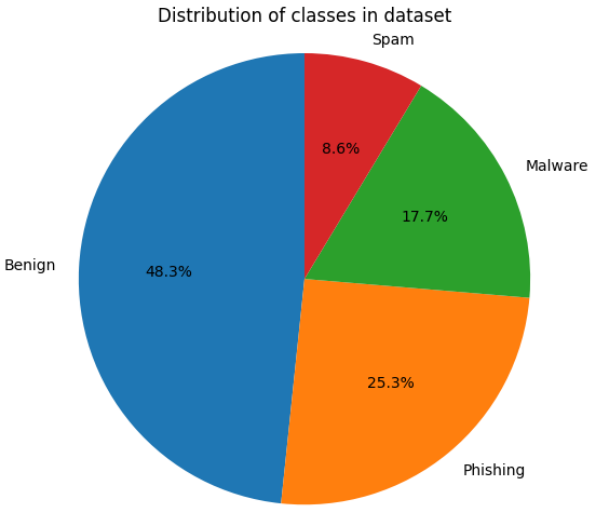

Figure 4: % Distribution of domain classes in the dataset

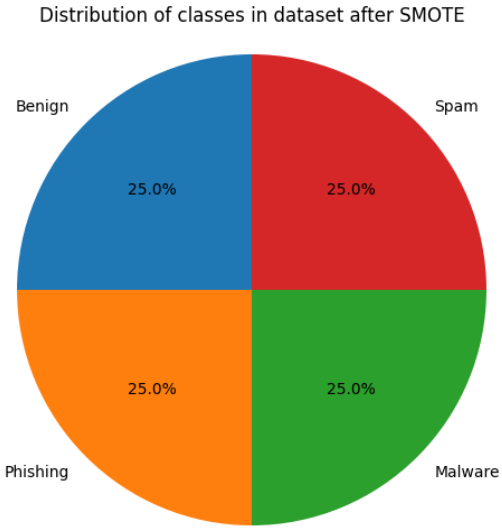

Figure 5: % Distribution of domain classes in the dataset after SMOTE

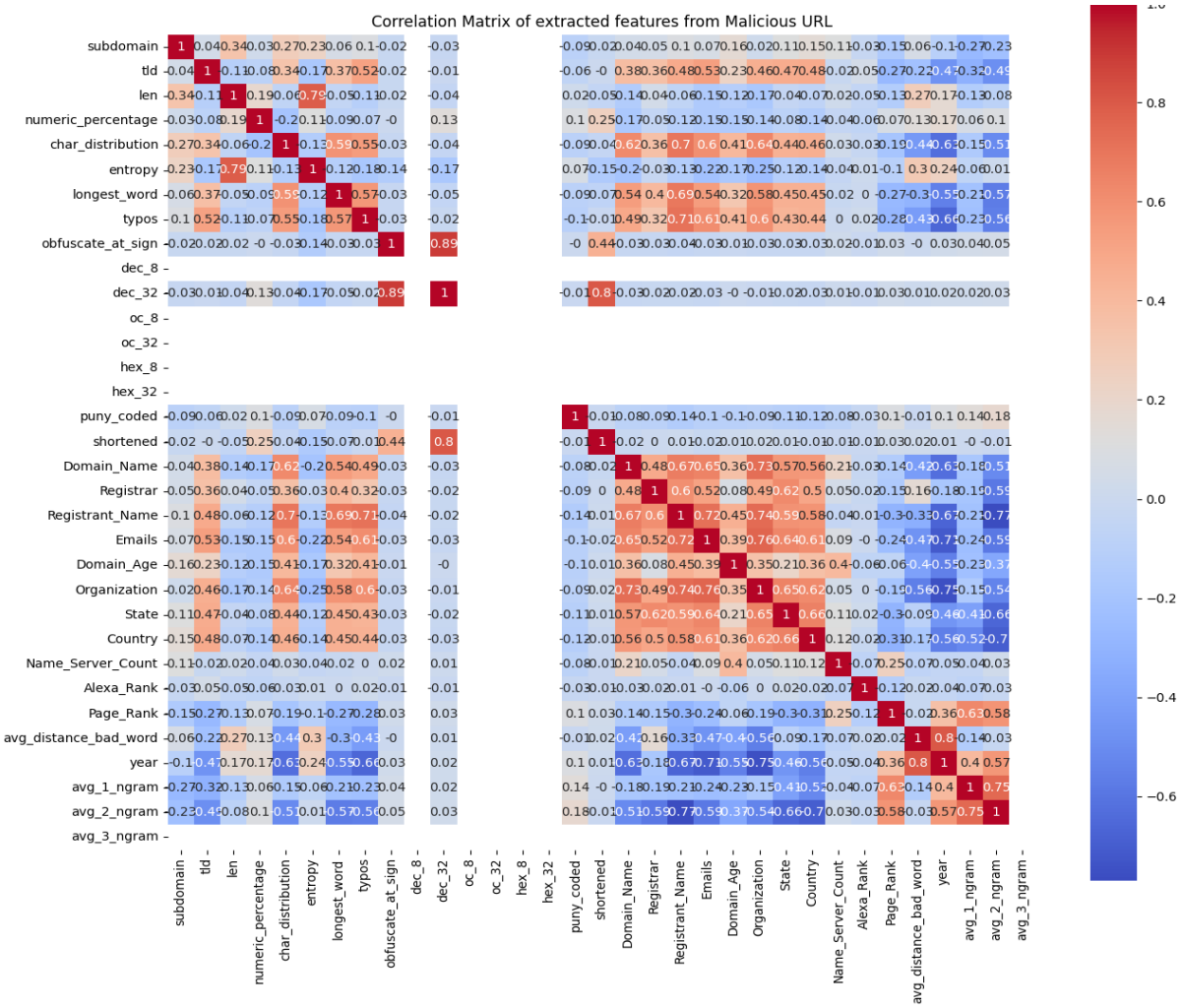

Figure 6: Correlation Matrix for dataset

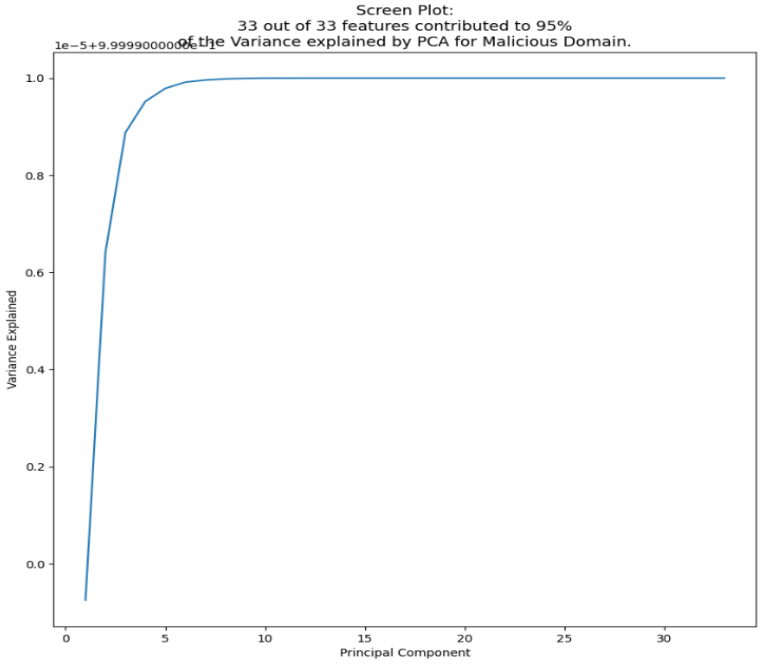

Figure 7: PCA diagram for dataset

	feature	VIF	Tolerance
19	Alexa_Rank	1.0	1.000000
9	puny_coded	1.1	0.909091
3	numeric_percentage	1.2	0.833333
0	subdomain	1.4	0.714286
20	Page_Rank	1.4	0.714286
18	Name_Server_Count	1.5	0.666667
1	tld	1.7	0.588235
14	Domain_Age	1.7	0.588235
12	Registrar	1.9	0.526316
6	longest_word	2.0	0.500000
7	typos	2.2	0.454545
17	Country	2.3	0.434783
4	char_distribution	2.5	0.400000
11	Domain_Name	2.7	0.370370
16	State	2.7	0.370370
8	dec_32	2.9	0.344828
10	shortened	3.0	0.333333
5	entropy	3.1	0.322581
2	len	3.2	0.312500
13	Emails	3.2	0.312500
15	Organization	3.8	0.263158

Figure 8: Feature selection by VIF

Table 5: Feature selected after Variance Inflation Factor Computation.

Feature type	Selected features after VIF
DNS statistical	None
Lexical Features	'subdomain', 'tld', 'len', 'numeric_percentage', 'char_distribution', 'entropy', 'longest_word', 'typos', 'dec_32', 'puny_coded', 'shortened
third-party	'Domain_Name', 'Registrar', 'Emails', 'Domain_Age', 'Organization', 'State', 'Country', 'Name_Server_Count', 'Alexa_Rank', 'Page_Rank'

Phase 2: Model Development, Evaluation, and Deployment.

A. Model Development and Evaluation

Model development was conducted in two phases. Initially, the models listed in Table 3 were trained using an 80/20 split ratio for training and testing. Subsequently, 5-fold and 10-fold cross-validation were employed to enhance model generalization.

Table 6 displays the performance evaluation of various machine learning models based on an 80/20 split ratio. This evaluation includes classification metrics such as accuracy, precision, recall, and F1-score, as well as regression metrics like Mean Squared Error (MSE) and Root Mean Squared Error (RMSE), along with the execution time of each model. Figures 9, 10, and 11 illustrate the classification metrics, while Figure 12 shows the RMSE values for each model corresponding to the 80/20 split ratio. Additionally, Figures 13 and 14 present the mean accuracy results for 5-fold and 10-fold cross-validation, respectively.

.

B. State-of-the-Art Model Comparison

The performance of the model was compared to other state-of-the-art approaches that utilized the public CICBellDNS2021 dataset to train AI/ML models. The results of this comparison are presented in Table 7. The findings indicate that the method used in this study surpassed the other state-of-the-art techniques, achieving a perfect accuracy score of 100% for Extreme Gradient Boosting (XGBoost), Random Forest (RForest), and Gradient Boosting (GRB) algorithms.

Table 6: Performance Evaluations of Models

	model	accuracy	r2	rmse	mse	preci	f1_score	recall	exe_time
0	RForest	1.000	1.000	0.017	0.000	1.000	1.000	1.000	1.815
1	Adaboost	0.958	0.947	0.255	0.065	0.959	0.959	0.959	1.059
2	GRB	1.000	1.000	0.000	0.000	1.000	1.000	1.000	16.719
3	MLP	0.867	0.773	0.528	0.279	0.871	0.865	0.869	5.441
4	KNN	0.911	0.859	0.417	0.174	0.913	0.912	0.911	0.883
5	LogReg	0.772	0.584	0.715	0.512	0.788	0.774	0.774	21.559
6	DT	0.998	0.998	0.045	0.002	0.998	0.998	0.998	0.092
7	SVM	0.843	0.724	0.583	0.340	0.851	0.845	0.844	0.614
8	XGBoost	1.000	1.000	0.017	0.000	1.000	1.000	1.000	0.444
9	NBayes	0.939	0.947	0.256	0.066	0.942	0.940	0.939	0.014

Table 7: Model comparison of state-of-the-art approaches that have used the CIC-Bell-DNS2021 dataset for training models

Author(s)	Model Applied	Accuracy & F1-score
Mahdavifar et al, [17]	KNN	Accuracy:98.9%, F1-Score: 98.9%
Kumaar et al., [45]	ImmuneNet	Accuracy: 99.2%, F1-Score: 99.2%
Zhao et al. [46]	(1DCNN)	Accuracy:95.6%, F1-Score: 94.8%
Egwali & Ekhator [47]	Random forest	Accuracy:89.9%, F1-Score:90.5%
Ma et al. [48]	BiLSTM	Accuracy: 92.38%
Maruthupandi et al.,[44]	Deep convoluted bi-LSTM network (AConBN)	Accuracy: 99.51%, F1-Score: 99.54%
Proposed system	XGBoost, RForest & GRB	Accuracy: 100%, F1-Score: 100%

Figure 9: Accuracy performance of models

Figure 10: Precision performance of models

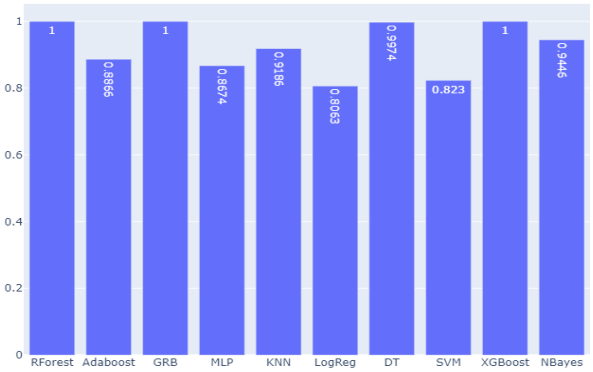

Figure 11: F1-score performance of models

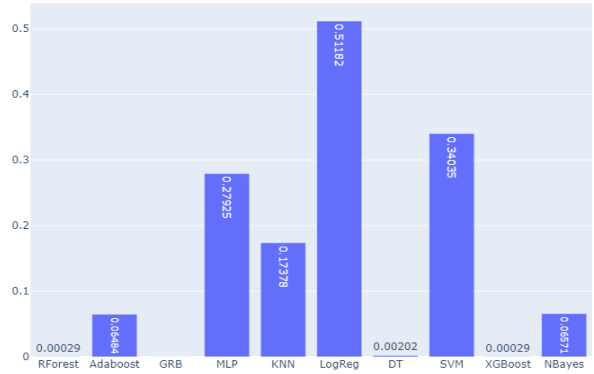

Figure 12: RMSE values for various models

Figure 13: Mean Accuracy of models for 5-fold cross-validation

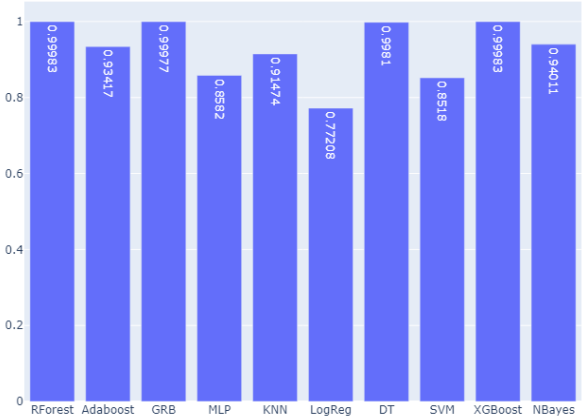

Figure 13: Mean Accuracy of models for 10-fold cross-validation

4.2 Discussion of Results

The approach presented in this study highlights the importance of extracting knowledge from databases or datasets. Evaluating the results of training advanced AI and machine learning (ML) models with balanced multiclass malicious domain name (MDN) datasets provides valuable insights that inform crucial decisions about classifying these MDNs. The tree-based AI algorithms, including Random Forest (RForest), Gradient Boosting (GRB), and Extreme Gradient Boosting (XGBoost), demonstrated outstanding performance based on classification metrics, achieving an accuracy score of 100%, precision score of 100%, recall score of 100%, and an F1-score of 100% as presented in Figures 9, 10 and 11 respectively. Additionally, the root mean square error (RMSE) for the tree-based AI models (RForest, GRB, and XGBoost) was below 0.00029 when compared to other ML models.

The mean accuracy results from both 5-fold and 10-fold cross-validation are shown in Figures 13 and 14, respectively. Empirical evidence indicates that XGBoost, Random Forest (RForest), and Gradient Boosting (GRB) produced superior results using the cross-validation training approach. This further supports the assertion that these tree-based AI models demonstrate enhanced generalization capabilities for the classification of multiclass and unknown MDNs.

We determined the best-performing model (BPM) by analyzing correlation and regression metrics, where both XGBoost and Random Forest (RForest) excelled. Additionally, we evaluated the execution time for classifying multiclass MDNs. With an execution time of 0.44 milliseconds, XGBoost emerged as the preferred model for real-time classification and identification of malicious domain names.

Additionally, when comparing these models with existing state-of-the-art approaches, XGBoost and RForest continue to show superiority. Although Kumaar et al. [45] and Maruthupandi et al. [44] achieved impressive accuracy results, their training and classification execution times were high due to the complexity of their model structures compared to tree-based AI models used in this study.

In light of the results presented, we seek to address the research questions mentioned earlier.

Question 1: What key features enhance the detection of malicious domain names across various categories?
This study was able to extract some key features that enhance the detection of MDNs across various categories of benign, spam, phishing, and malware as presented by the VIF feature selection presented in Figure 8.

Question 2: How effective are machine learning algorithms in identifying malicious domain names used for spamming, distributing malware, and conducting phishing attacks?
The results in Table 6 provide evidence that AI/ML models can effectively classify malicious domains used for spamming, phishing, and malware attacks based on extracted DNS features.

Question 3: Which machine learning algorithms can deliver outstanding performance for a multi-class MDN classification system?
The results from Tables 6 and 7 demonstrate the effectiveness of tree-based algorithms such as Random Forest and Extreme Gradient Boosting.

Both models exhibited exceptional performance in the multiclass MDN classification system.

5. Conclusion

The emergence of AI-powered malicious domain names has rendered traditional classification methods ineffective. To tackle this issue, it is essential to employ advanced AI and machine learning algorithms in conjunction with a refined data mining strategy that enhances the predictive capability of models. This study underscores the significance of maintaining a balanced dataset, as well as utilizing techniques such as dimensionality reduction and variance inflation factor imputation in the development of a multiclass malicious domain name detection system. An 80/20 split ratio was initially applied for model training, and a cross-validation approach was also implemented to promote model generalization. Performance metrics for both classification and regression were extracted during model training and evaluation.

The results demonstrate that tree-based AI and machine learning models, such as XGBoost and Random Forest, outperformed other models. Notably, XGBoost exhibited the fastest processing time while achieving the highest performance.

Future research should focus on experimenting with additional datasets used in Maruthupandi et al. [44] to evaluate the performance of XGBoost across various datasets. Additionally, it would be beneficial to implement the best-performing model as a web browser extension for real-time detection. For example, we could develop plugins that collect URLs from browser histories, extract domain names, and classify them. This approach is crucial, as many malicious domain name attacks occur through web browsers to gain control over network resources.

Declaration of interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding This work did not receive any funding from individuals, organizations, or government.

References

[1]. Falowo, O. I., Ozer, M., Li, C., & Abdo, J. B. (2024). Evolving Malware and DDoS Attacks: Decadal Longitudinal Study. IEEE Access, 12, 39221–39237. https://doi.org/10.1109/ACCESS.2024.3376682

[2]. Chaisse, J., & Friedmann, D. (2023). Law of the Digital Domain: Trademarks, Domain Names, and the AI Frontier. *IDEA*, *64*, 399.

[3]. Ali, M. L., Thakur, K., & Obaidat, M. A. (2024). The Application of Layered Authentication in Cybersecurity. In *2024 International Conference on Computing, Internet of Things and Microwave Systems (ICCIMS)*, 1-5, IEEE.

[4]. Nashikkar, R. N., Padhye, Y. N., & Ingle, R. (2023, December). Enhancing Malicious Domain Detection Using Advanced Machine Learning Techniques. In *2023 IEEE Pune Section International Conference (PuneCon)*, 1-6, IEEE.

[5]. Ayeni, R. K., Adebiyi, A. A., Okesola, J. O., & Igekele, E. (2024). Phishing Attacks and Detection Techniques: A Systematic Review. In *2024 International Conference on Science, Engineering, and Business for Driving Sustainable Development Goals (SEB4SDG)*, 1-17, IEEE.

[6]. Saputra, H., Stiawan, D., & Satria, H. (2023). Malware Detection in Portable Document Format (PDF) Files with Byte Frequency Distribution (BFD) and Support Vector Machine (SVM). *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, *9*(4), 1144-1153.

[7]. Madhvan, R., & Zolkipli, M. F. (2023). An Overview of Malware Injection Attacks: Techniques, Impacts, and Countermeasures. *Borneo International Journal eISSN 2636-9826*, *6*(3), 22-30.

[8]. Rus, A. C., El-Hajj, M., & Sarmah, D. K. (2024). NAISS: A reverse proxy approach to mitigate MageCart's e-skimmers in e-commerce. *Computers & Security*, *140*, 103797.

[9]. Phung, N. M., & Mimura, M. (2024). Malicious JavaScript Detection in Realistic Environments with SVM and MLP Models. *Journal of Information Processing*, *32*, 748-756.

[10]. Karajgar, M. D., Sawardekar, S., Khamankar, S., Tiwari, N., Patil, M., Borate, V. K., ... & Chaudhari, A. (2024). Comparison of Machine Learning Models for Identifying Malicious URLs. In *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)*, 1-5, IEEE.

[11]. Rodríguez, E., Anghel, R., Parkin, S., Van Eeten, M., & Gañán, C. (2023). Two Sides of the Shield: Understanding Protective {DNS} adoption factors. In *32nd USENIX Security Symposium (USENIX Security 23)*, 3135-3152.

[12]. Gladin, I., Ewards, V., & Terence, S. (2024). Domain Name Server Filtering Service Using Threat Intelligence and Machine Learning Techniques. In *International Conference on Inventive Communication and Computational Technologies*, 529-540, Singapore: Springer Nature Singapore.

[13]. Abu Al-Haija, Q., Alohaly, M., & Odeh, A. (2023). A lightweight double-stage scheme to identify malicious DNS over HTTPS traffic using a hybrid learning approach. *Sensors*, *23*(7), 3489.

[14]. Ayub, M. A., Smith, S., Siraj, A., & Tinker, P. (2021). Domain Generating Algorithm based Malicious Domains Detection. In *2021 8th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2021 7th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, 77-82, IEEE.

[15]. Behnke, M., Briner, N., Cullen, D., Schwerdtfeger, K., Warren, J., Basnet, R., & Doleck, T. (2021). Feature engineering and machine learning model comparison for malicious activity detection in the dns-over-https protocol. *IEEE Access*, *9*, 129902-129916.

[16]. Carr, A., Alam, A., & Allison, J. (2023). Monitoring Malicious DNS Queries: An Experimental Case Study of Utilising the National Cyber Security Centre's Protective DNS within a UK Public Sector Organisation.

[17]. Mahdavifar, S., Maleki, N., Lashkari, A. H., Broda, M., & Razavi, A. H. (2021). Classifying malicious domains using DNS traffic analysis. In *2021 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)* 60-67, IEEE.

[18]. Demmese, F. A. (2023). *Machine Learning Based Traffic Classification Using Image Visualization* (Doctoral dissertation, North Carolina Agricultural and Technical State University).

[19]. Ma, S., Pang, T., Cui, R., & Yang, D. (2024). A Malicious Domain Detection Method Based on DNS Logs. In *2024 4th International Conference on Blockchain Technology and Information Security (ICBCTIS)*, 283-288, IEEE.

[20]. Panigrahi, G. R., Sethy, P. K., Behera, S. K., Gupta, M., Alenizi, F. A., Suanpang, P., & Nanthaamornphong, A. (2024). Analytical Validation and Integration of CIC-Bell-DNS-EXF-2021 Dataset on Security Information & Event Management. *IEEE Access*.

[21]. Singhdeo, T. J., Singhdeo, A., Mohanty, J. R., & Satapathy, S. (2023). Artificial Cognitive Intelligence and Information Technology in Cybersecurity. In *International Conference on Computer & Communication Technologies*, 347-354, Singapore, Springer Nature.

[22]. Wang, H., Tang, Z., Li, H., Zhang, J., & Cai, C. (2023). DDOFM: Dynamic malicious domain detection method based on feature mining. *Computers & Security*, *130*, 103260.

[23]. Mankar, N. P., Sakunde, P. E., Zurange, S., Date, A., Borate, V., & Mali, Y. K. (2024, April). Comparative Evaluation of Machine Learning Models for Malicious URL Detection. In *2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon)*, 1-7, IEEE.

[24]. Park, K. H., Song, H. M., Do Yoo, J., Hong, S. Y., Cho, B., Kim, K., & Kim, H. K. (2022). Unsupervised malicious domain detection with less labeling effort. *Computers & Security*, *116*, 102662.

[25]. Gomez, G., Kotzias, P., Dell'Amico, M., Bilge, L., & Caballero, J. (2023). Unsupervised detection and clustering of malicious tls flows. *Security and Communication Networks*, *2023*(1), 3676692.

[26]. Liao, R., & Wang, S. (2024). Malicious domain detection based on semi-supervised learning and parameter optimization. *IET Communications*, *18*(6), 386-397.

[27]. Zhang, J., Sun, H., & Wang, J. (2022). Malicious domain name detection model based on CNN-LSTM. In *Third International Conference on Computer Communication and Network Security (CCNS 2022)*, 12453, 57-62, SPIE.

[28]. Ke, W., Zheng, D., Zhang, C., Deng, B., Yao, H., & Tian, L. (2022). CGFMD: CNN and GRU Based Framework for Malicious Domain Name Detection. In *International Conference on Artificial Intelligence and Securit*, 564-574, Cham: Springer International Publishing.

[29]. Aarthi, B., Jeenath Shafana, N., Flavia, J., & Chelliah, B. J. (2022). A hybrid multiclass classifier approach for the detection of malicious domain names using RNN model. In *Computational Vision and Bio-Inspired Computing: Proceedings of ICCVBIC 2021*, 471-482, Singapore: Springer.

[30]. Zhang, X., Wang, C., Liu, R., & Yang, S. (2024). Federated rnn-based detection of ransomware attacks: A privacy-preserving approach.

[31]. Pradeepa, G., & Devi, R. (2022). Malicious domain detection using nlp methods—a review. In *2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART)*, 1584-1588, IEEE.

[32]. Balasubaramanian, S., Ganesan, P., & Rajasekaran, J. (2023). Weighted ensemble classifier for malicious link detection using natural language processing. *International Journal of Pervasive Computing and Communications*.

[33]. Hamroun, C., Amamou, A., Haddadou, K., Haroun, H., & Pujolle, G. (2024). A review on lexical based malicious domain name detection methods. *Annals of Telecommunications*, 1-17.

[34]. Rozi, M. F., Ozawa, S., Ban, T., Kim, S., Takahashi, T., & Inoue, D. (2022). Understanding the influence of AST-JS for improving malicious webpage detection. *Applied Sciences*, *12*(24), 12916.

[35]. Yang, L., Liu, G., Wang, J., Zhai, J., & Dai, Y. (2022). A semantic element representation model for malicious domain name detection. *Journal of Information Security and Applications*, *66*, 103148.

[36]. Manasrah, A. M., Khdour, T., & Freehat, R. (2022). DGA-based botnets detection using DNS traffic mining. *Journal of King Saud University-Computer and Information Sciences*, *34*(5), 2045-2061.

[37]. Nowroozi, E., Mohammadi, M., & Conti, M. (2022). An adversarial attack analysis on malicious advertisement URL detection framework. *IEEE Transactions on Network and Service Management*, *20*(2), 1332-1344.

[38]. He, D., Dai, J., Gu, H., Zhu, S., Chan, S., Su, J., & Guizani, M. (2022). A Malicious Domains Detection Method Based on File Sandbox Traffic. *IEEE Network*.

[39]. GHEORGHIȚĂ, C. A., SMADA, D., VEVERA, A. V., DUMITRACHE, M., SANDU, I. E., & ROTUNĂ, C. I. (2023). Blacklists and whitelists in the framework of a domain reputation system. *Romanian Journal of Information Technology & Automatic Control/Revista Română de Informatică și Automatică*, *33*(4).

[40]. Bayer, J., Maroofi, S., Hureau, O., Duda, A., & Korczynski, M. (2023). Building a Resilient Domain Whitelist to Enhance Phishing Blacklist Accuracy. In *2023 APWG Symposium on Electronic Crime Research (eCrime)*, 1-14. IEEE.

[41]. Sachan, R. K., Agarwal, R., & Shukla, S. K. (2023). Identifying malicious accounts in blockchains using domain names and associated temporal properties. *Blockchain: Research and Applications*, *4*(3), 100136.

[42]. Wang, Q., Dong, C., Jian, S., Du, D., Lu, Z., Qi, Y., ... & Liu, Y. (2023). HANDOM: Heterogeneous attention network model for malicious domain detection. *Computers & Security*, *125*, 103059.

[43]. Gao, Y., Li, Z., Yuan, F., Zhang, X., Wang, D., Cao, C., & Liu, Y. (2023). Robust Malicious Domain Detection Against Adversarial Attacks on Heterogeneous Graph. In *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2028-2033, IEEE.

[44]. Maruthupandi, J., Sivakumar, S., Dhevi, B. L., Prasanna, S., Priya, R. K., & Selvarajan, S. (2025). An intelligent attention based deep convoluted learning (IADCL) model for smart healthcare security. *Scientific Reports*, *15*(1), 1363.

[45]. Kumaar, A. M., Samiayya, D., Vincent, P. D. R., Srinivasan, K., Chang, C. Y., & Ganesh, H. (2022). A hybrid framework for intrusion detection in healthcare systems using deep learning. *Frontiers in Public Health*, *9*, 824898

[46]. Zhao, H., Han, L., & Wang, W. (2023). Detection of COVID-19-related Malicious Domain Names Based on Feature Fusion. In *2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, 1251-1256, IEEE.

[47]. Egwali, A.O, & Ekhator R.O. (2024) Malicious Domain Names Detection using Deep-Learning Classifiers, *Journal of Computing, Science &Technology*, 1(1), 81-86, https://focjournal.unidel.edu.ng/

[48]. Ma, S., Pang, T., Cui, R., & Yang, D. (2024). A Malicious Domain Detection Method Based on DNS Logs. In *2024 4th International Conference on Blockchain Technology and Information Security (ICBCTIS)*, 283-288, IEEE

[49]. Streun, C. A. (2024). *Detecting Modern Maldocs: An Analytical Study of PDF Feature Importance Using KDD Cup 99 as the Process Framework*. The George Washington University.

[50]. Savenko, B., Lysenko, S., Bobrovnikova, K., Savenko, O., & Markowsky, G. (2021). Detection DNS tunneling botnets. In *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, 1, 64-69, IEEE.