Journal of Theoretical and Applied Information Technology
15th July 2025. Vol.103. No.13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MALICIOUS DOMAIN DETECTION USING INTEGRATED SUPERVISED AND UNSUPERVISED MACHINE LEARNING APPROACHES

SRI LAXMI KUNA [1], PULLURI SRINIVAS RAO[2], A. LAKSHMANARAO[3], SOWNDARYA LAHARI CHINTALAPUDI[4], NALANAGULA HARINI[5], HARI KRISHNA H[6]

[1]Associate Professor, Department of IT, MVSR Engineering College, Hyderabad, Telangana, India
[2]Professor, Department of CSE, Jayamukhi Institute of Technological Sciences,Narsampet,Warangal, India
[3]Department of IT, Aditya University, Surampalem, India
[4]Department of CSE (AI&ML, DS), Anil Neerukonda Institute of Technology & Sciences, Sangivalsa,Visakhapatnam, India
[5]Assistant Professor, Department of CSE, Vignan's Institute of Engineering for Women, India
[6]Assistant Professor, Department of CSE, Ballari Institute of Technology and Management, Ballari, India
E-mail: [1]drsrilaxmi2019@gmail.com, [2]srithanrao@gmail.com, [3]a.lakshmanarao@adityauniversity.in, [4]sowndaryatekkali@gmail.com, [5]harinii.nalan23@gmaicom, [6]harivanam87@gmail.com

ABSTRACT

Detecting Domain Generation Algorithms (DGA) is crucial in cybersecurity to identify malicious domain names. While existing studies focus individually on either supervised or unsupervised learning, limited work has explored their integrated use for DGA detection. This paper addresses that gap by combining clustering-derived features with traditional classifiers to enhance detection accuracy. This paper explores an innovative approach for DGA detection utilizing supervised classification and unsupervised clustering techniques. The methodology begins with preprocessing the dataset and extracting relevant features, such as domain names, host information, and subclass labels. Later, feature hashing is utilized for dimensionality reduction, transforming categorical features like domain names, hosts, and subclasses into feature vectors. Advanced clustering methods, including KMeans, Hierarchical Clustering (Agglomerative), and Density-Based Clustering (DBSCAN), are employed to uncover underlying patterns in the data. These techniques aid in identifying distinct groups or clusters within the dataset, potentially assisting in differentiating DGA from legitimate domain names. Later, cluster labels were added as features for final dataset. Subsequently, multiple ML classifiers, including Random Forest, Decision Tree, KNN, SVM, and Logistic Regression, are trained to classify domain names as DGA or non-DGA based on the extracted features. Rigorous experimentation and evaluation assess the performance of each classifier in terms of accuracy and other relevant metrics. This hybrid approach contributes new knowledge on how feature enrichment through clustering can improve model generalization in real-world cyber threat scenarios. The results offer insights into the effectiveness of the proposed methodologies for DGA detection.

Keywords: *DGA Detection, Supervised learning, Unsupervised Learning, Machine Learning, Random Forest.*

1. INTRODUCTION

In the dynamic landscape of cybersecurity, combating the ever-evolving threats posed by malicious entities remains a top priority for organizations and security professionals worldwide. Amongst these threats, DGAs are a major concern due to their enabling role in malwares and botnets nefarious actions. DGAs are an advanced mechanism used by cybercriminals to bury the trail of their malicious activity while evading traditional security. With an ever-growing dependence on the digital infrastructure, there are still new cyberattacks that targets the weaknesses of the domain name system (DNS), robust and adaptive detection systems are more and more crucial. The dynamism of DGAs precludes the use of simple blacklist-based approaches, which requires more intelligent techniques and systems that can catch the complex patterns.

In contrast to static domain names that can be easily identified and blocked, DGA-generated domains are constantly changing, which makes it and difficult

for security defenses. Accordingly, there is an urgent requirement for alternative methods that can continue to accommodate this changeable threat, and successfully identify and mitigate DGAs.

Over the past few years, the boundary between supervised and unsupervised learning approaches has been coming together as the solution to finding hidden patterns in "large scale" sets of data. These techniques are not only automated with feature extraction but also generalized to new and unseen DGA variants. This bi-way approach improves detection accuracy by leveraging the pros of both classification and clustering.

One promising direction for DGA detection is the use of machine learning and clustering. Security experts can now use more sophisticated algorithms and data driven approaches to discover subtle patterns and anomalies in large datasets of domain names. By these means, the underlying structures and relationships are explored to find signs that may signal the existence of the DGA generated domains, through which the security practitioners could detect and mitigate the threats in advance.

This paper proposes an integrated framework that leverages both supervised and unsupervised learning models to detect malicious domains generated by DGAs. The method is aimed to provide robust and resource efficient testing on the one hand and elimination of the need of costly computation (such as machine learning) for checking on the other, by augmenting classical classifiers py the cluster features. This approach can help model the connection between theoretical detection and practical deployment in cybersecurity systems.

Although progress has been achieved in DGA detection under supervised and unsupervised single methodologies, there is still no end-to-end solution that integrates both types of methods together, which also makes full use of the complementary advantages of them. This study is important in that it fills the gap by demonstrating the potential gain of detection performance by incorporating features derived from clustering with conventional classifiers. This hybrid approach benefits from better classification performance as well as offering a scalable and adaptable technology for actual application to cybersecurity systems.

2. LITERATURE SURVEY

In [1], the authors proposed a method for detecting DGA-generated domain names, focusing specifically on non-English, particularly Chinese, domain names composed of Pinyin. They adopted the strategy of dividing domain names into subwords to achieve contextual representation with the FastText model. A deep learning model was used to categorize the vectorized domain names either as DGA-generated or as benign. In [2], the authors have focused on the challenge of DNS resolution, which has become a major problem in term of latency affecting user's experience. In order to avoid improper solutions (DNS resolution performance enhancement through complex, expensive architectural deployments…) or heavy user device embedded components (integrated modules), a user-centric predictive resolution decision approach is proposed. This technique is designed to refine DNS responses, especially in edge infrastructures where computation resource is scarce, enhance quality of access and reduce resolution time for end users. In [3], authors presented improvements to mitigate the inaccuracy of detecting malicious domains, especially with respect to short DNS communication data obtained from malware infected terminals. They added access time information to the responses and transformed domain querying into primary toplevel domain names, overcoming previous constraints. The experimental results showed that such modifications helped to improve the detection accuracy on malicious domains. In [4], the authors analyzed the practice of internal domain naming in home networks via active measurements (RIPE Atlas), while prior works used passive measurements. 34.51% of which were potentially subject to collision, which resolved to a total of 3,092 internal names served by 4,305 probes.

In [5], the authors undertook a longitudinal analysis of domain name registrations associated with high profile global events, with the Olympics being used as an example of registrant motivation, use, possible abuse etc. Analyzing the Tokyo, Beijing and Paris Olympics during the 3 years, they found that there were a large number of ODN registrations in conjunction with the rescheduling of the 2020 Tokyo Olympics and the 2022 Beijing Olympics' diplomatic boycott. In [6], researchers investigated deep learning (DL) systems that combined NLP with (MLP) to enhancing the detection of malicious domain registrations. In [7], the authors handle the false positives in DGA

detection tools given non-English domain names, especially in the context of Chinese domains. They proposed a detection method that includes a domain name embedding approach to capture linguistic patterns effectively. In [8], the authors proposed a transfer learning-based named entity recognition model that combines lite BERT, Bi-GRU, and CRF to enhance automatic entity acquisition across domains. The model uses ALBERT for character vector generation and Bi-GRU to capture contextual relationships. Experimental results showed improved F1 scores, demonstrating the model's effectiveness in cross-domain applications. In [9], the authors compared deep learning models for Named Entity Recognition, RNN, LSTM, GRU, and CNN. The study revealed significant accuracy improvements across models, highlighting their strengths and limitations for future NER research. In [10], the authors investigate the misuse of ChatGPT-related squatting domains in cybersecurity, identifying over 1.3 million such domains through a novel method leveraging historical Passive DNS data.

In [11], the authors analyze the risks associated with Internationalized Domain Names (IDNs), focusing on the IDN homograph problem. They develop the IDNMon framework for a large-scale measurement study involving 863 top-level domain zone files and historical data. The authors in [12] noted that existing DGA domain name detection methods often failed due to easily evaded features. They proposed a two-stage feature reinforcement method, which encoded domain names into word vectors and utilized a slice pyramid network for feature extraction. By integrating semantic information and reducing redundancy, they improved feature stability.The authors in [13] proposed a feature fusion method for detecting COVID-19-related malicious domain names, utilizing WHOIS data and features extracted with a Transformer and 1DCNN. Their method achieved good accuracy and precision, proving effective in identifying malicious domains. The authors in [14] studied DGA domain name characteristics and proposed a CNN-BiLSTM detection model that extracts local and global features. Their experimental results showed significant improvements in detection performance.

[15] investigated the vulnerability of the DNS to abuse and drew attention to the power of deep learning for detecting attack patterns. They recognized the difficulty of existing research works and suggested more efficient methods for the detection of single-character or word concatenated DNS domains, where the sensitive events are fundamental in identifying the malicious patterns. Authors in [16] introduced a new algorithm to identify malicious domain names conformed on statistical features of URL characters. They used the relevant features and built a decision tree to analyse such characters which could identify in-de-marcs-exist-ing and generated domain names. The accuracy and precision were of ≈90%, which showed an excellent performance in the detection of malicious domain. The work in [17] compared features of malicious domain names and natural ones, presenting the weaknesses of DNS to a range of attacks. They extracted features of malicious DNS using Python and two types of feature categories to support Internationalized Domain Names. The detection of DGA domain names represent a significant risk in network security as they are used in multiple types of attacks and have been targeted in previous efforts [18]. Traditional deep learning models were good in automatic feature extraction but could not do well on the wordlist-base DGA domains. They addressed this by adding semantic features in addition to character features to enhance the accuracy. The authors of [19] were concerned with the problem of malicious domain name identification which is aimed at promoting people’s privacy and possessions. They high level idiots have recommended a "new" detection algorithm based on statistical observations of URL characters. The works in [20] aimed at improving the detection of illegal web pages for Internet security. They also proposed a data pre-processing model and applied it on the largest one passive DNS database. Hybrid-temporary random domain name filtering algorithm Based on the technique of LSTM-CNN for filtering illegal web-page and the combination of features for speed-up and the enhancement of accuracy an illegal web-page detection algorithm was designed. In [21], the authors proposed a lightweight DL model for counterfeit domain name detection using only domain name strings to address the issue of feature extraction in machine learning.

The authors have focused on detecting malicious domain name in [22], and introduced a new solution which combines knowledge graph and DNS information. They utilized a wrapper based approach to construct a DNS information knowledge graph by fusing DNS flow graph and DNS domain name hierarchy graph together. The model jointly learns both the entities and the

attributes and, it vectorizes them using the knowledge graph and the NN. The work in [23] focused on the botnet DGA domain name detection under ransomware, and they proposed an RL-LSTM model which is used to create new DGA samples based on a small amount of real samples. This model employs reinforcement learning to improve the domain name generation ability of LSTM. [24] evaluated 13 state-of-the-art classification methods by using the same dataset, which includes 80 DGA families, and performed an experimental comparison to test the detection performance of new DGA-generated malicious domain names. They used 3 methods to extract text features – unigram, bigram, and trigram, and achieved decent results for each ML model.

This paper not only develops upon existing techniques for identifying DGA-generated domain, but also provides valuable experience of using unsupervised clustering into the supervised learning to detect malicious domains. Comparing classification accuracy with and without clustering-derived features demonstrates the practical effect of data structuring on classification measures. Readers will gain insight into the role of hybrid learning techniques in the improvement of cybersecurity tools useful for future adaptive threat detection systems.

3. RESEARCH METHODOLOGY

The framework of the proposed DGA detection method is illustrated in Fig. 1. Initial stages: In the proposed methodolo- gies, the process starts by collecting a mixed dataset, which com- bines domain information, host detail and its sub-class labels, to detect DGAs that are related to malware and botnet. After collecting data, there is some preprocessing, which includes cleaning and normalizing domain names as well as removing noise and encoding categories. Then, more advanced clustering algorithms including KMeans, Hierarchical Clustering (Agglomerative) and Density-Based Clustering (DBSCAN) are used to expose and explore hidden patterns within the data. Feature hashing is used to convert categorical features to fixed-length feature vectors for dimension reduction and enabling more effective clustering. Features of interest are then extracted and engineered to capture vital signs of DGA activity. Several standard ML classifiers such as Random Forest, Decision Tree, KNN, SVM, and Logistic Regression are next fitted on the pre-processed data. The performance of each classifier for distinguishing the domain names between DGA and non-DGA are evaluated using standard metrics including accuracy, precision, recall and F1-score. Extensive experiments and cross-validation are performed to demonstrate the efficiency of the developed techniques on parameter tuning and fine-tuning of algorithms to maximize the detection accuracy. Finally, we also analyze the results from the experiments to understand the robustness and weakness of the detection framework, which serves as a reference for the research in the security field.

3.1. Collection of Data and Preprocessing

The data acquisition phase involved sourcing a comprehensive dataset from Kaggle, comprising domain names, host information, and subclass labels, crucial for detecting Domain Generation Algorithms (DGAs) associated with malware and botnets. The dataset, with a shape of (160,000, 4), provided a diverse set of samples for analysis.

Prior to analysis, the obtained dataset experienced intense standardization during the preprocessing step. The “hashing trick” (Luminata et al., 2017) and feature hashing (also known as the hashing trick) were used for the domain and subclass features. This processing could convert these features to a set of fixed-length feature vectors, and then reduce the dimensionality of the feature vectors and improve clustering performance. In addition, one-hot encoding was applied to just the "subclass" feature as it contained only nine unique values _. This encoding could easily transform as a binary matrix format which could be better suitable to machine learning algorithms for the representation of subclasses.

3.2. Unsupervised Learning Algorithms for Cluster Analysis

Advanced clustering methods including KMeans, Hierarchical Clustering (Agglomerative) and Density-Based Clustering (DBSCAN) were applied during the clustering analysis process to identify intrinsic patterns and structures of data were conducted.

Journal of Theoretical and Applied Information Technology
15th July 2025. Vol.103. No.13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

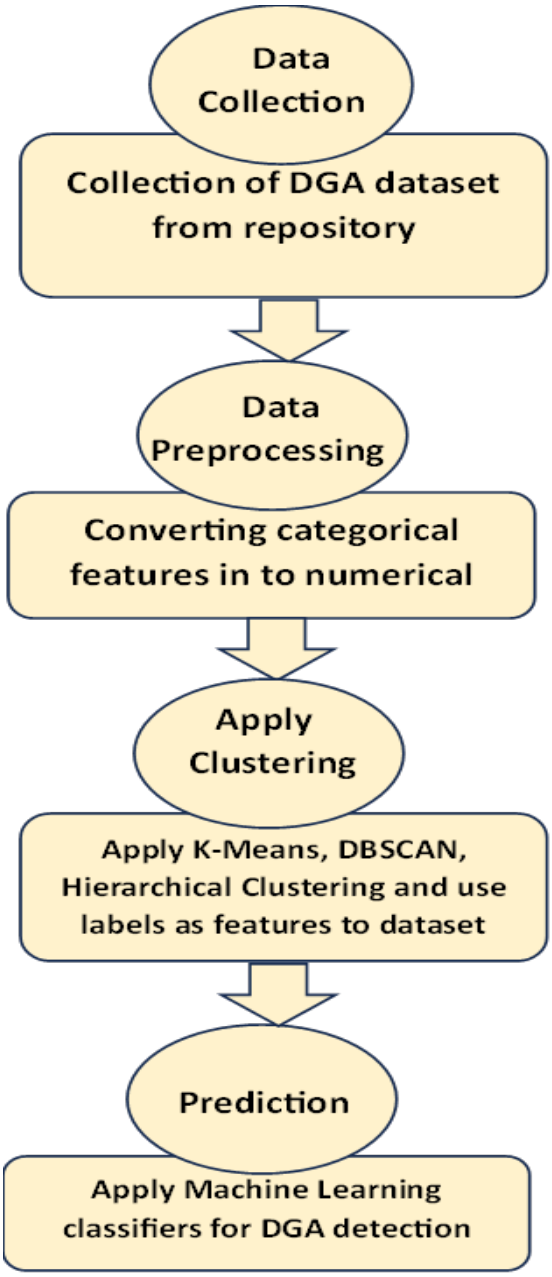

Figure 1: Proposed Method for lung cancer detection

Unsupervised Clustering Algorithms are crucial for identifying patterns as signals of malicious domain generation. All such algorithms have unique techniques for performing pairwise clustering of domains based on their features. KMeans clustering divides the data points into k clusters that minimize variance among clusters, where it separates so called similar domains. Agglomerative Hierarchical Clustering produces a dendrogram that is used to visualize the hierarchical relationships between the domains. This hierarchical method of clustering aids in the interpretation of a data set's complex structure by allowing the identification of clusters at the most relevant levels of granularity. Likewise, DBSCAN also finds the dense region in data space where the definition of cluster is defined as a high density connected points therefore it is capable of finding clusters of arbitrary shapes and size. These

clustering approaches provide an interesting overview of the domain data structure and are suitable for identifying the abnormal and malicious activities.

Aggregate the labels after applying the clustering algorithms in the next stage of application of py ML. Through the inclusion of these cluster labels as features, the Machine Learning models can incorporate learned patterns and structures from the preliminary clustering phase to better detect and classify malicious activity associated domains. This fusion of clustering and Machine Learning innovations sets a foundation for holistic DGA detection and cybersecurity analysis.

3.3. Supervised Machine Learning Algorithms

In the machine learning classification phase, a number of algorithms were used to classify domain names into DGA or non-DGA. Random Forest, a powerful ensemble learning algorithm, builds multiple decision trees in the training process to vote for classes to solve the problem of large-scale datasets which are highly dimensional. A Decision Tree classifier repeatedly split the dataset into subsets according to feature values, where the process iterates, building to a tree of decisions, making the model easier to interpret and visualize. For example, K-Nearest Neighbors (KNN), a non-parametric algorithm, categorizes the sample according to the most frequent class among the k closest. Support Vector Machine (SVM) is a popular supervised learning algorithm that can be used to generate hyperplanes that can be used to separate different classes with maximum margin, suitable in high-dimensional spaces, and capable of using kernel functions to match different class boundary. Logistic Regression is a linear model for binary classification that outputs the log odds of the probability of the original probability of a sample belonging to a class as the response, but has many other uses because of its simplicity and interpretability. With the promote of this combinatory classifier, the performance and generalization of the DGA detection system was adequately and effectively tested and improved.

4. EXPERIMENTS AND RESULTS

4.1 Applying K-Means Clustering

K-Means clustering involves an iterative process which assigns the data points to the nearest cluster and after each step it recalculates the cluster centers with the mean of all the assigned points to that cluster and it continues the next step until the assignment doesn't change. Clustering data into groups makes it easier for you to explore its patterns and structure, helping you make better decisions and predictions. The elbow method which is shown on Figure 2 gives you an idea on the impact of the number of clusters (k) on the within-cluster sum of squares. This technique can help choose the best value of k for KMeans clustering. By drawing a plot of the number of clusters versus the within-cluster sum of squares, the sum of squares does not keep sliding smoothly down as the number of clusters grows. But there comes a point when the decline flattens out sharply, to form a kind of distinct "elbow" in the plot. This elbow point is thought to be the number of clusters in the dataset. By visualizing the plot, the 'elbow' breaking point clear could be seen, which indicated that the dataset was best clustered into 4 labels. The optimal number of cluster has to trade off the amount of variance that is captured against overfitting. So, let's go ahead and perform KMeans clustering with 4 clusters to divide the data nicely.

4.2 Applying DBSCAN Clustering

Similarly, for DBSCAN clustering, a comparable approach was adopted to ascertain the optimal number of clusters. The optimal number of clusters was determined to be 5 by the number clusters quality trade-off plot across min_samples for a quality measure such as silhouette score or Davies-Bouldin index. It is about measuring the fit of data points to the clusters that they have been assigned, a high score here meaning that the clusters are well separated. The most coherent and meaningful clustering solution was identified by stepping the parameter min_samples and examining the impact on cluster quality metrics. Choosing five clusters resulted in the best compromise for representing different clusters of data having members with some degree of similarity. This careful attention to details results in clusters that are true to the true underlying structure of the dataset, and allow for useful interpretations and actionable insights.

Journal of Theoretical and Applied Information Technology
15th July 2025. Vol.103. No.13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.3 Applying Hierarchical Clustering

With the use of dendrograms from Hierarchical Clustering (Agglomerative), a 4-cluster solution was formed as shown in Figure 3. This included the manipulation of dendrogram to represent the hierarchical relationships between the points and investigation of the clusters for various linkage distances. The structure of the dendrogram was analysed to find natural breakpoints/levels of clustering, which most accurately described the inherent organization of the data. The hierarchical methods used gave good insight into patterns and structures of data, which could be further analyzed and interpreted. When partitioning the dataset into 4 clusters, a trade-off between granularity and internal consistency was achieved - the pattern in each cluster was distinctive, and internally coherent. Following clustering using KMeans, Hierarchical Clustering (Agglomerative), and Density-Based Clustering (DBSCAN), the resulting cluster labels were added as new features in the dataset. These cluster labels were informative as to the underlying groupings and trends present in the data. Enriching the dataset with this membership information as features of the samples transformed the datasets to contain information on how the samples were members of different clusters. This expansion of the data set with cluster labels enabled an expanded representation of the underlying patterns and regularities of the data. It also gave machine learning algorithms more data to distinguish between class or group differences in the dataset.

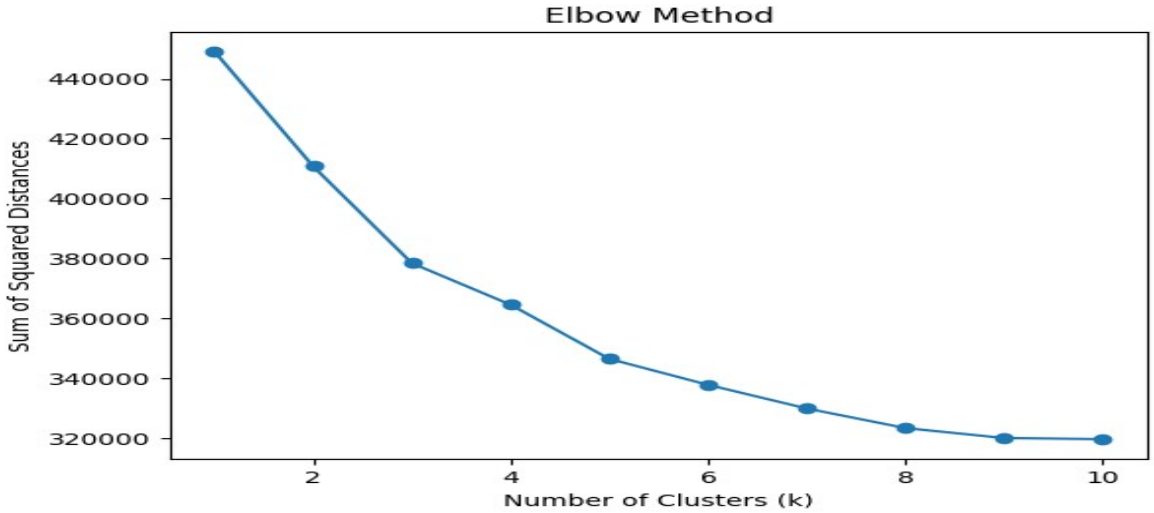

Figure 2: Elbow method graph

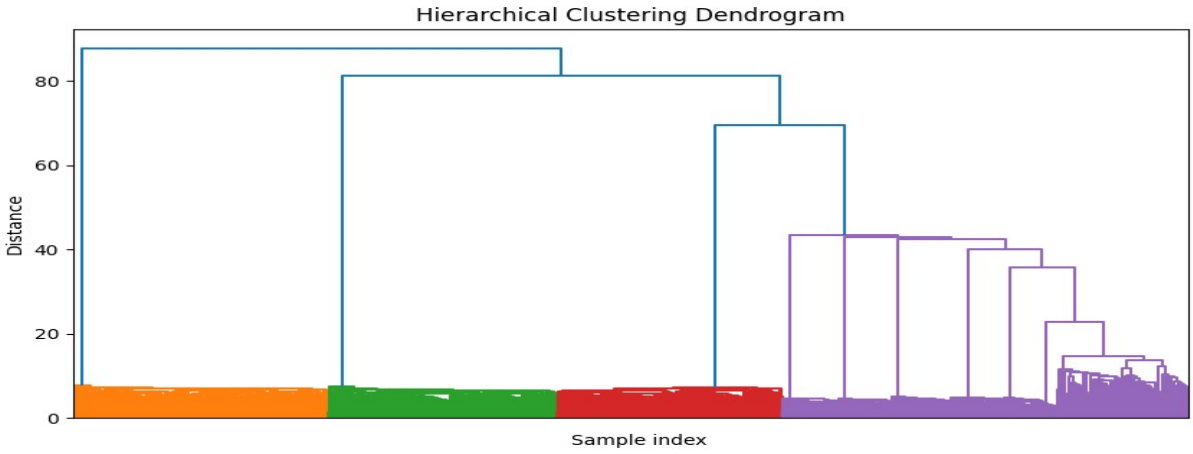

Figure 3: Dendrogram for Hierarchical Clustering

Table 1: Performance of ML methods

Algorithm	Accuracy	F1 score
Random Forest	99.50%	99.00%
Decision Tree	98.50%	98%
KNN	97%	96.70%
SVM	95.50%	95%
Logistic Regression	96%	96%

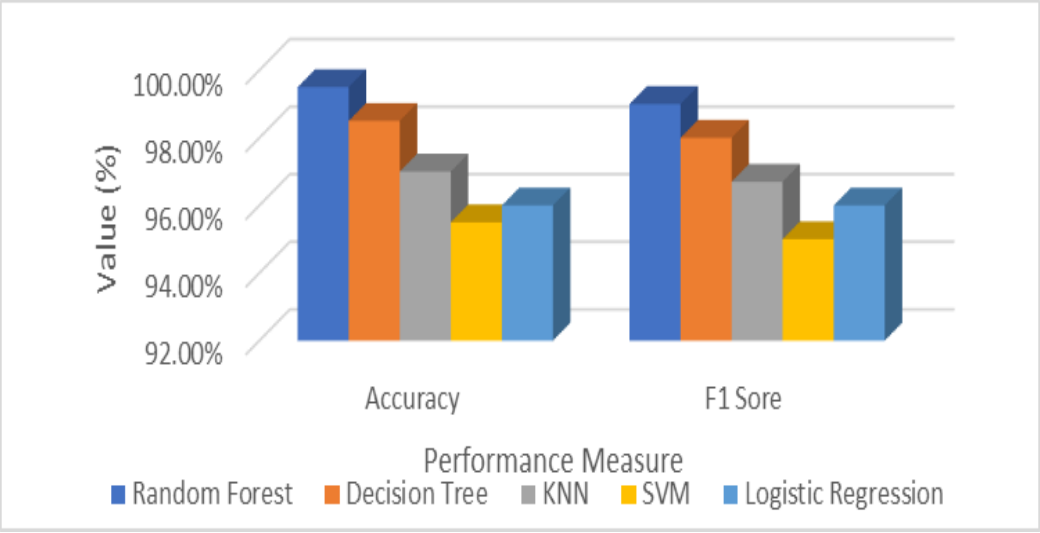

Figure 4: Supervised Learning algorithms results

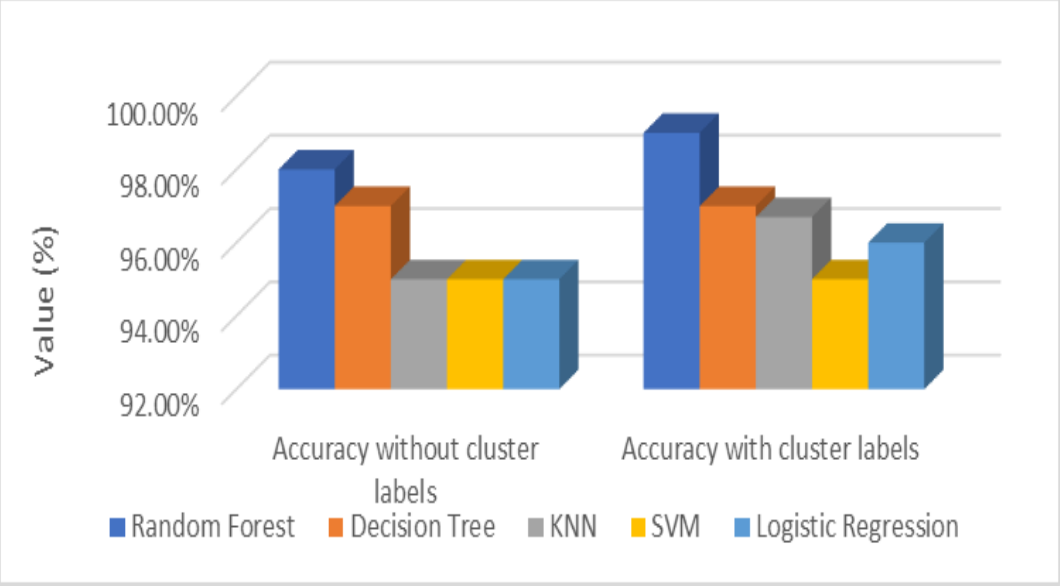

Figure 5: Model Performance with and without Cluster Labels

Table 2: Model Performance with and without Cluster Labels

Algorithm	Accuracy without cluster labels	Accuracy with cluster labels
Random Forest	98.00%	99.00%
Decision Tree	97.00%	97.00%
KNN	95%	96.70%
SVM	95.00%	95%
Logistic Regression	95%	96%

4.4 Applying Supervised Learning algorithms

Table 1, Figure 4show the performance measures of different supervised ML algorithms after including the cluster labels as additional features. In particular, Random Forest model had the highest accuracy of 99.50% and the best F1 score of 99.00%. This means that training the model with the cluster labels has lead to a much better calibration of instances. Next one is the Decision Tree model with the accuracy of 98.50%, F1 score of 98%.

Random Forest and Decision Tree models both exhibit strong performance, using the cluster labels to improve predictive capability better than random labels. The KNN algorithm also had reasonable performance with this combined cluster labels (97% accuracy, 96.7% F1 score). Likewise, the SVM model achieved a strong 95.5 and 95% accuracy and F1 score. Logistic Regression model with cluster labels resulted in an accuracy of 96% and F1 score of 96% eventually.

The performance of machine learning models with and without using cluster labels as additional features was shown using Table 2 and Fig 5. The Random Forest model had an accuracy of 98% prior to integration of cluster labels, but the accuracy improved to 99% when cluster labels were added. Likewise, the Decision Tree model remainted with 97% of accuracy with and without the cluster labels. Especially, for the KNN algorithm, accuracy increased from 95% without cluster labels to 96.7% with cluster labels, which suggests the favorable role of integrating cluster information for classification

SVM and Logistic Regression models have an accuracy of 95.00%, both without cluster labels, which improved to 96.00% for both models when cluster labels are included. This demonstrates the advantage of using cluster labels for improving the predictability of these models. In summary, these results underscore the importance of the cluster labels as auxiliary features that enhance performance of machine learning models for classification problems.

Although the presented method achieves higher accuracy by clustering and classification, it has some weaknesses. Depending on cluster quality is a problem however, since noisy or poorly separated clusters may inject ambiguity into the feature set. Second, the method has not been developed and tested on real-time streaming data or multilingual domain patterns, which restricts its application in more general cybersecurity environments. These are the areas which we feel can be further improved and experimented.

4.5 Comparison with Prior Work

Contrary to the existing works which purely focused on either supervised or unsupervised learning for DGA generated domain detection, this study combines them to design a hybrid detection framework. In the supervised learning process, cluster labels are used as additional features and the proposed approach achieves an improvement in detection accuracy. The integration is important, since a complementary research question revolves around integrating clustering and classification for cybersecurity applications, which is still an underinvestigated area in the literature. The test results have shown that combining those structural insights from clustering with classification models results in more robust and generalizable detection performances, which is the main purpose of this work.

5. CONCLUSION

This work provides a method for DGAs detection in cyber threat using the combination of machine learning and cluster analysis. Underlying patterns in the dataset were discovered by using advanced clustering techniques like K Means, Hierarchical Clustering and Density Based Clustering to split malicious DGAs from legitimate domain names. Feature hashing was used to reduce feature dimension and transform categorical features into fixed-size feature vectors. After another round of training, a number of machine learning classifiers could achieve exciting scores of categorizing domain names into DGA or non-DGA according to the generated features, including Random Forest (RF), Decision Tree (DT), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Logistic Regression (LR). The integration of cluster labels as common features has remarkably enhanced performance of machine learning classifiers in terms of the accuracy rate for all algorithms. In particular, Random Forests and Decision Trees exhibited substantial accuracy gains upon incorporating cluster labels. One can see that Random Forest has its accuracy increased from 98.00% to 99.00%, it's accuracy remain the same as Decision Tree with 97.00%. It's also noteworthy that KNN and Logistic Regression classifiers showed a promising performance improvement of utilizing cluster labels in addition to the original feature values for DGA detection, confirming that our approach could be employed to achieve better DGA detection accuracy. The proposed technique contributes to the literature by empirically confirming the performance gain that the integration between unsupervised clustering features and supervised learning-based classifiers bring—they have been little explored within the DGA detection literature to the best of our knowledge. The main technical contribution of this paper is in illustrating how unsupervised cluster features can be successfully incorporated into supervised models in order to improve the identification of DGA-generated malicious domains. This hybrid approach extends the current body of cybersecurity research by providing a scalable accurate and generalizable detection methodology.

REFERENCES:

[1] H. Lee, J. Do Yoo, S. Jeong and H. K. Kim, "Detecting Domain Names Generated by DGAs With Low False Positives in Chinese Domain Names," in IEEE Access, vol. 12, pp. 123716-123730, 2024, doi: 10.1109/ACCESS.2024.3454242.

[2] I. F. Ferreira and E. Oki, "Latency-Aware Cache Mechanism for Resolver Service of Domain Name Systems," NOMS 2024-2024 IEEE Network Operations and Management Symposium, Seoul, Korea, Republic of, 2024, pp. 1-4, doi: 10.1109/NOMS59830.2024.10575387.

[3] T. Koga, D. Nobayashi and T. Ikenaga, "Accuracy Improvement Method for Malicious Domain Detection Using Machine Learning," 2024 IEEE 21st Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 2024, pp. 1108-1109, doi: 10.1109/CCNC51664.2024.10454674.

[4] E. Boswell and C. Perkins, "RIPEn at Home - Surveying Internal Domain Names Using RIPE Atlas," 2024 8th Network Traffic Measurement and Analysis Conference (TMA), Dresden, Germany, 2024, pp. 1-4, doi: 10.23919/TMA62044.2024.10559012..

[5] R. Kawaoka, D. Chiba, T. Watanabe, M. Akiyama and T. Mori, "Longitudinal Measurement Study of the Domain Names Associated With the Olympic Games," in IEEE Access, vol. 12, pp. 19128-19144, 2024, doi: 10.1109/ACCESS.2024.3360108.

[6] F. Çolhak, M. İ. Ecevit, H. Dağ and R. Creutzburg, "Comparing Deep Neural Networks and Machine Learning for Detecting Malicious Domain Name Registrations," 2024 IEEE International Conference on Omni-layer Intelligent Systems (COINS), London, United Kingdom, 2024, pp. 1-4, doi: 10.1109/COINS61597.2024.10622643.

[7] H. Lee and H. K. Kim, "Mitigating False Positives in DGA Detection for Non-English Domain Names," 2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks - Supplemental Volume (DSN-S), Brisbane, Australia, 2024, pp. 150-151, doi: 10.1109/DSN-S60304.2024.00042.

[8] L. Qingyu and Z. Gang, "Cross-domain big data named entity recognition based on transfer learning," 2024 IEEE 4th International Conference on Electronic Technology, Communication and Information (ICETCI), Changchun, China, 2024, pp. 1340-1344, doi: 10.1109/ICETCI61221.2024.10594555.

[9] K. S, P. S. M, P. C and M. K, "Enhancing Named Entity Recognition using Deep Learning Approaches," 2024 5th International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2024, pp. 1733-1737, doi: 10.1109/ICESC60852.2024.10690015.

[10] M. Liu et al., "ChatScam: Unveiling the Rising Impact of ChatGPT on Domain Name Abuse," 2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Brisbane, Australia, 2024, pp. 507-521, doi: 10.1109/DSN58291.2024.00055.

[11] Y. Zhang et al., "Understanding and Characterizing the Adoption of Internationalized Domain Names in Practice," in IEEE Transactions on Dependable and Secure Computing, doi: 10.1109/TDSC.2024.3386905.

[12] H. Yang, T. Zhang, Z. Hu, L. Zhang and X. Cheng, "A DGA Domain Name Detection Method Based on Two-Stage Feature Reinforcement," 2023 IEEE 22nd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Exeter, United Kingdom, 2023.

[13] H. Zhao, L. Han and W. Wang, "Detection of COVID-19-related Malicious Domain Names Based on Feature Fusion," 2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Rio de Janeiro, Brazil, 2023, pp. 1251-1256, doi: 10.1109/CSCWD57460.2023.10152588.

[14] Y. Wang, R. Pan, Z. Wang and L. Li, "A Classification Method Based on CNN-BiLSTM for Difficult Detecting DGA Domain Name," 2023 IEEE 13th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 2023, pp. 17-21, doi: 10.1109/ICEIEC58029.2023.10200702.

[15] L. Zhu and J. Nie, "Deep Learning Approaches to Detecting Malicious Domains in DNS: Challenges and Opportunities," 2023 IEEE 11th International Conference on Information, Communication and Networks (ICICN), Xi'an, China, 2023, pp. 858-861, doi: 10.1109/ICICN59530.2023.10393096.

[16] H. Zhao, Z. Chen and R. Yan, "Malicious Domain Names Detection Algorithm Based on Statistical Features of URLs," 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Hangzhou, China, 2022, pp. 11-16, doi: 10.1109/CSCWD54268.2022.9776264.

[17] A. Almarzooqi, J. Mahmoud, B. Alzaabi, A. Ghebremichael and M. Aldwairi, "Detecting Malicious Domains Using Statistical Internationalized Domain Name Features in Top Level Domains," 2022 14th Annual Undergraduate Research Conference on Applied Computing (URC), Dubai, United Arab Emirates, 2022, pp. 1-6, doi: 10.1109/URC58160.2022.10054226.

[18] R. Pan, J. Chen, H. Ma and X. Bai, "Using Extended Character Feature in Bi-LSTM for DGA Domain Name Detection," 2022 IEEE/ACIS 22nd International Conference on Computer and Information Science (ICIS), Zhuhai, China, 2022, pp. 115-118, doi: 10.1109/ICIS54925.2022.9882343.

[19] H. Zhao, Z. Chen and R. Yan, "Malicious Domain Names Detection Algorithm Based on Statistical Features of URLs," 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Hangzhou, China, 2022, pp. 11-16, doi: 10.1109/CSCWD54268.2022.9776264.

[20] Y. Su, B. Peng and X. Li, "Fast Illegal Webpage Detection Algorithm Based on Massive Domain Name Resolution Records," 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, 2022, pp. 4313-4322, doi: 10.1109/BigData55660.2022.10020782.

[21] Z. Wang and W. Yang, "Deep Learning-based Algorithm for Detecting Counterfeit Domain Names," 2022 7th International Conference on Multimedia Communication Technologies (ICMCT), Xiamen, China, 2022, pp. 60-65, doi: 10.1109/ICMCT57031.2022.00020.

[22] Z. Dong, X. Chen, J. Zhao, S. Zhao and J. Wu, "Malicious Domain Name Detection based on Knowledge Graph," 2022 IEEE International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), Chongqing, China, 2022, pp. 251-256, doi: 10.1109/SDPC55702.2022.9915824.

[23] H. Cheng, Y. Fang, L. Chen and J. Cai, "Detecting Domain Generation Algorithms Based on Reinforcement Learning," 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Guilin, China, 2019, pp. 261-264, doi: 10.1109/CyberC.2019.00051.

[24] M. M. Rayhan and M. A. Ayub, "An Experimental Analysis of Classification Techniques for Domain Generating Algorithms (DGA) based Malicious Domains Detection," 2020 23rd International Conference on Computer and Information Technology (ICCIT), DHAKA, Bangladesh, 2020, pp. 1-5, doi: 10.1109/ICCIT51783.2020.9392701.