Machine Learning-based Information Security Model for Botnet Detection

Heba M. Fadhil, Noor Q. Makhool, Muna M. Hummady, Zinah O. Dawood

Department of Information and Communication, Al-Khwarizmi College of Engineering, University of Baghdad, Iraq

Emails: heba@kecbu.uobaghdad.edu.iq; noor.q@kecbu.uobaghdad.edu.iq; muna@kecbu.uobaghdad.edu.iq; zina_osama@kecbu.uobaghdad.edu.iq

Abstract

Botnet detection develops a challenging problem in numerous fields such as order, cybersecurity, law, finance, healthcare, and so on. The botnet signifies the group of co-operated Internet connected devices controlled by cyber criminals for starting co-ordinated attacks and applying various malicious events. While the botnet is seamlessly dynamic with developing counter-measures projected by both network and host-based detection techniques, the convention techniques are failed to attain sufficient safety to botnet threats. Thus, machine learning approaches are established for detecting and classifying botnets for cybersecurity. This article presents a novel dragonfly algorithm with multi-class support vector machines enabled botnet detection for information security. For effectual recognition of botnets, the proposed model involves data pre-processing at the initial stage. Besides, the model is utilized for the identification and classification of botnets that exist in the network. In order to optimally adjust the SVM parameters, the DFA is utilized and consequently resulting in enhanced outcomes. The presented model has the ability in accomplishing improved botnet detection performance. A wide-ranging experimental analysis is performed and the results are inspected under several aspects. The experimental results indicated the efficiency of our model over existing methods.

Keywords: Botnet detection; Machine learning; Data Classification; MSVM model; Dragonfly algorithm

1. Introduction

A network is an assortment of gadgets that are associated with the Internet, and the complete number of these gadgets expands consistently. Without a doubt, organizations of monetary and business foundations are constantly under security risk that not just costs billions of dollars in harm and recuperation, yet in addition, negatively affects their standing. The rising number of clients impacted by pernicious programming is turning into a basic issue. Botnets have turned into the fundamental worry since they are probably the greatest danger to security frameworks. Its prevalence comes in its capacity to control corporate centralized computers by invading any Internet-associated gadget that utilizes an advanced video recorder (DVR) [1]. The botnet is characterized as an organization of compromised have gadgets that are utilized to complete pernicious exercises [2]. A botnet comprises of three parts: an assailant called a botmaster, order and control (C&C) server, and a tainted machine named as bot. The botmaster requires a C&C channel to order the bots and direction vindictive assaults [3]. Fig. 1 depicts the process in Botnet detection and classification.

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

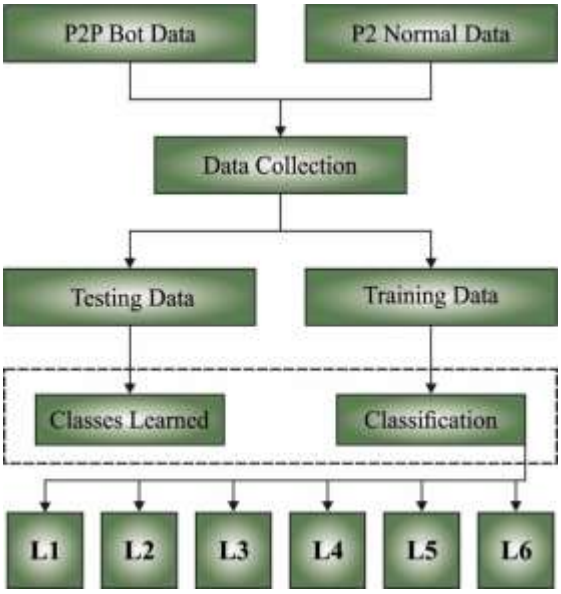

Figure 1: Process in Botnet Detection and Classification

Late exploration works on botnets among our studied writing centers for the most part around planning frameworks to distinguish order and control (C and C) botnets, where numerous bot-tainted machines are controlled and facilitated by couple of substances to complete malevolent exercises [4, 5]. Consequently, ML based classifiers are at the center of individuals frameworks and are regularly prepared by named information in regulated learning conditions [6]. The most well-known classifier is SVMs with various parts, while spatial-fleeting time series investigation and probabilistic inductions are additionally remarkable procedures utilized in Machine learning-based classifiers. Bunching is for the most part utilized in regular language handling (NLP), to assemble a huge scope framework to distinguish bot questions [7, 8]. The ground truth is regularly heuristic or a mix, marked by human specialists, for instance, the game experts' visual investigations distinguish bots in web based games [9]. By and large, the development of botnet recognition is obvious from prior and more direct purposes of arrangement strategies, for example, bunching and Naive Bayes, the exploration center has been extended from the last advance of characterization to the significant going before step of building reasonable measurements, which gauges and recognizes bot-based and human-based exercises [10].

The researchers in [11] projected a graph-based ML technique for botnet recognition that first considers the consequence of graph feature beforehand emerging a generalized method for botnet detection according to the certain significant feature. Then, explored distinct feature subsets chosen by means of five filter-based feature assessment measures acquired from different concepts namely information, consistency, and correlation. Popoola et al. [12] presented an effective DL-based botnet attack recognition method which could manage extremely imbalanced network traffics. Especially, Synthetic Minority Oversampling Method (SMOTE) creates further minority models to accomplish class balance, whereas Deep RNN (DRNN) learns hierarchical feature representation from the balanced network traffics to implement discriminatory classification. Ashraf et al. [13] developed a statistical learning-based botnet architecture, named IoTBoT-IDS, that secures IoT-based smart networks against botnet attacks. IoTBoT-IDS capture the standard behavior of IoT network through statistical learning-based methods, with a Correntropy model and Beta Mixture Model (BMM). Ibrahim et al. [14] present a multi-layer architecture for detecting botnets through machine learning algorithm that consists of classification and filtering modules to distinguish the control server and botnet command. Then, emphasized various conditions for this architecture, especially it should be protocol-independent, structure-independent, and capable of detecting botnets in encapsulated method. Muhammad et al. [15] developed a technique for earlier-phase botnet recognition. The presented technique initially choses the optimum feature with feature selection technique. Then, it feeds this feature to ML classifier to estimate the accuracy of the botnet recognition.

This article presents a novel dragonfly algorithm with multi-class support vector machines that enabled botnet detection (DFA-MSVM) for information security. For effectual recognition of botnets, the DFA-MSVM model involves data pre-processing at the initial stage. Besides, the MSVM technique was utilized for the identification and classification of botnets that exist in the network. In order to optimally

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

adjust the SVM parameters, the DFA is utilized and consequently resulting in enhanced outcomes. The presented DFA-MSVM model has the ability in accomplishing improved botnet detection performance. A wide-ranging experimental analysis is performed and the results are inspected under several aspects.

2. The Proposed DFA-MSVM Model

In this article, a novel DFA-MSVM model has been developed for recognition and classification of botnet for information security. For effectual recognition of botnets, the DFA-MSVM model involves data pre-processing at the initial stage. Besides, the MSVM technique was utilized for the identification and classification of botnets that exist in the network. In order to optimally adjust the SVM parameters, the DFA is utilized and consequently resulting in enhanced outcomes.

2.1 Pre-processing

For constructing a bot detection structure, it can create classic social profile objects with social networks. An object infrastructure of profile is a method that has gradually utilized features of social media user accounts. The web crawler with various social network APIs was utilized to gather actual data set in open source data from OMSN user accounts. This approach can comprise problems due to its massive count of data, but, it significantly improves the time needed to classify profile objects as legitimate or infested profiles.

2.2 MSVM based Classification

In this work, the MSVM method was utilized for the identification and classification of botnets that exist in the network. The MSVM classifier depends on Vapnik–Chervonenkis (VC) dimension of statistical learning model [16]. A main function of MSVM is for mapping the pre-processing, nonlinear inseparable microarray gene expression data as to linear very dimension manifold θ with utilizes of alteration $\emptyset: R^N \to \theta$, then obtaining a better hyperplane: $\Psi: \psi(x) = (\omega \cdot \phi(x) + b)$ with solving the succeeding optimized convex problem (the soft margin problem):

$$\min(\omega, \xi) = \frac{1}{2}\omega^2 + \beta \sum_{i=1}^{n} \xi_i$$

Subjected to

$$y_i(\omega \cdot \phi(x) + b) \geq 1 - \xi_i, for\ all\ 1 \leq i \leq n, \quad (1)$$

whereas ω denotes the co-efficient vector of hyperplane in the manifold (feature space), b refers to the threshold value of hyperplane, ξ_i signifies the slack problem projected to classify error, and β denotes the penalty aspect to error. The higher value of β usually leads to a smaller margin which minimizes classification error, then lesser value of β is creates a wider margin resulting in several misclassified.

The common kernel function that is utilized as a continuous predictor, comprises as:

The linear kernel is demonstrated as:

$$G(x_i, x_{i'}) = x_i \cdot x_{i'}. \quad (2)$$

Afterward, the polynomial kernel is demonstrated utilizing in Eq. (28):

$$G(x_i, x_{i'}) = (\eta * (x_i \cdot x_{i'}) + \delta)^d, \quad (3)$$

whereas $> 0, \delta \in R$, and $d \in Z^+$.

At that time, the Gaussian kernel is formulated as:

$$G(x_i, x_{i'}) = \exp\left(\frac{x_i - x_i^2}{2\sigma^2}\right), \quad (4)$$

In which $\sigma > 0$.

This MSVM kernel function has approximately regarded as follows: local and global kernel functions [17]. Samples are very distinct have an enormous effect on the global kernel value then instanced neighboring everyone considerably controls the local kernel values.

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

2.3 DFA based Parameter Optimization

In order to optimally adjust the SVM parameters, the DFA is utilized and consequently resulting in enhanced outcomes. Lately, DFA technique is a projected bio inspired optimization method. Mostly, it is innovative from dynamic and static efficacy of dragonflies swarming behaviors. The DFA technique is composed of Hunting (static swarm) and Migration (dynamic swarm). In the event of static swarm, dragonflies create small group and move backward within a preferred location for hunting the victim. A flying feature of static swarm is instantaneous change and local movement. Therefore, the feature diverges from dynamic swarm, as large dragonfly, quantity of migrating on large distances form the swarm to transfer in one-direction. It is grownup from emergent sub-swarm and flies on numerous areas in a dynamic swarm that focuses on exploration phase wherever static swarm tries to transfer in bigger swarm as definer in exploitation phase [18].

The survival rate is the major goal of swarm where every individual is disseminated outside and attracted to the food location. Because of this behavior, 5 important features that influence the individual location are Alignment, Attraction, Cohesion, Separation to food source, and Distraction of enemy. Such characteristics are defined in the following.

Separation: This parameter is estimated by:

$$S_i = -\sum_{k=1}^{M} Y - Y_k, \tag{5}$$

Here Y signifies the specific location, Y_k defines the place of *k-th* neighbor individual and M defines the overall quantity of neighbor separations.

Alignment: It displays the mean of velocity that is defined as follows:

$$A_i = \frac{\sum_{k=1}^{M} V_k}{M}, \tag{6}$$

Now $V_k$characterize the velocity of *k-th* neighbor individual. Fig. 2 represents the flowchart of DFA [19].

Cohesion: here it is evaluated as follows:

$$C_i = \frac{\sum_{k=1}^{M} Y_k}{M} - Y \tag{7}$$

Attraction towards a food source: It characterizes a distance among position of existing individual and location of food source (Y^+):

$$F_i = Y^+ - Y \tag{8}$$

Distraction outwards an enemy: It indicates a distance among position of existing individual and position of enemy (Y^-):

$$E_i = Y^- - Y \tag{9}$$

In dragonfly, nature is a combination of 5 variables. Following, 2 vectors are employed for upgrading dragonfly location in a searching region, namely position vector (Y) and step vector (ΔY). It is formulated in the following:

$$\Delta Y_{t+1} = (aA_i + sS_i + cC_i + eE_i + fF_i) + w \Delta Y_t\,, \tag{10}$$

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

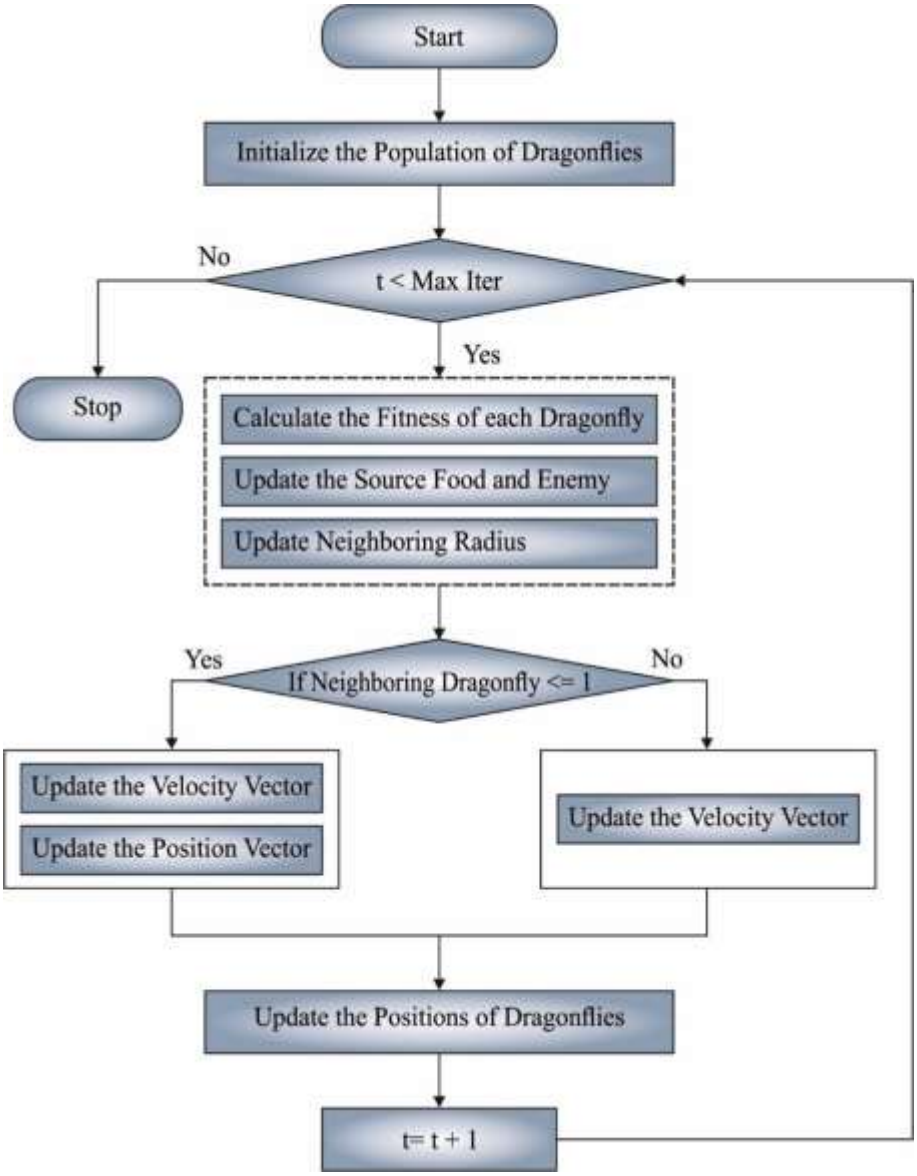

Figure 2: Flowchart of DFA

Now $A_i$signifies the alignment of *i-th* individual, a characterize the alignment weight, $S_i$shows the variance of *i-th* individuals, s denotes the separation weight, $C_i$symbolizes the cohesion of *i-th* individuals, c defines the cohesion weight, E_i describes the place of enemy in *i-th* individuals, e represents enemy weight, F_i shows a food source of *i-th* individuals, f specifies the food weight, t displays the iteration number and w show the inertia weight:

$$Y_{t+1} = Y_t + \Delta\, Y_{t+1} \tag{11}$$

During optimization technique, divergent explorative and exploitative behaviors are accomplished by a, s, c, e, and f parameters. Furthermore, they are employed in management of exploitation and exploration phases.

3. Performance Validation

In this section, the botnet detection and classification outcomes of the DFA-MSVM model are investigated using two datasets namely Facebook and Flickr.

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

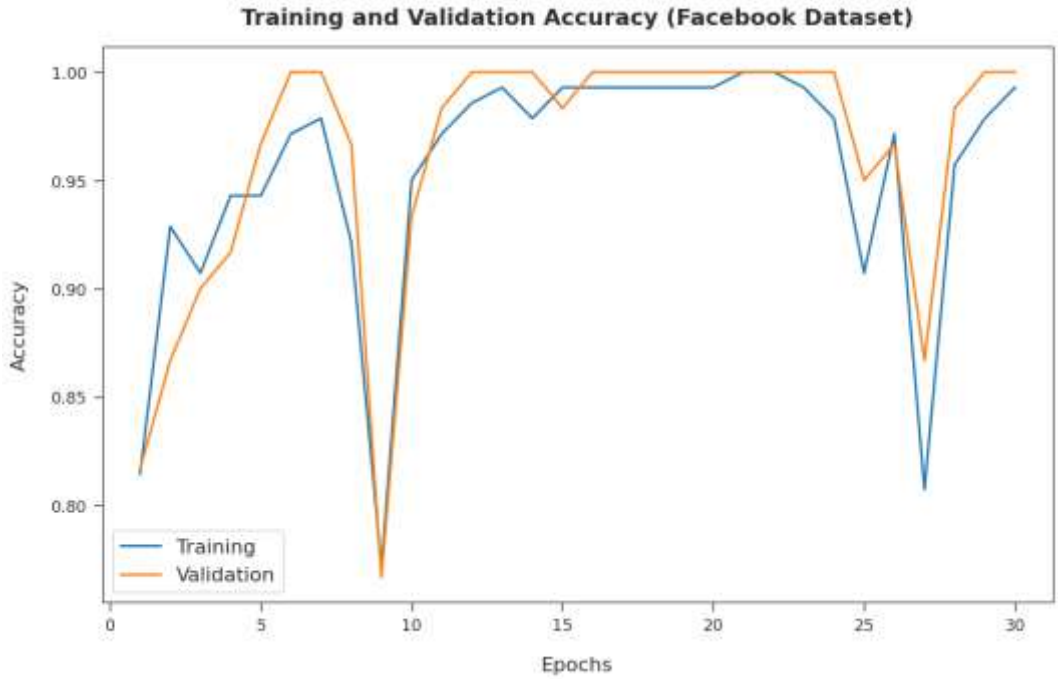

Figure 3: Accuracy analysis of DFA-MSVM technique on Facebook dataset

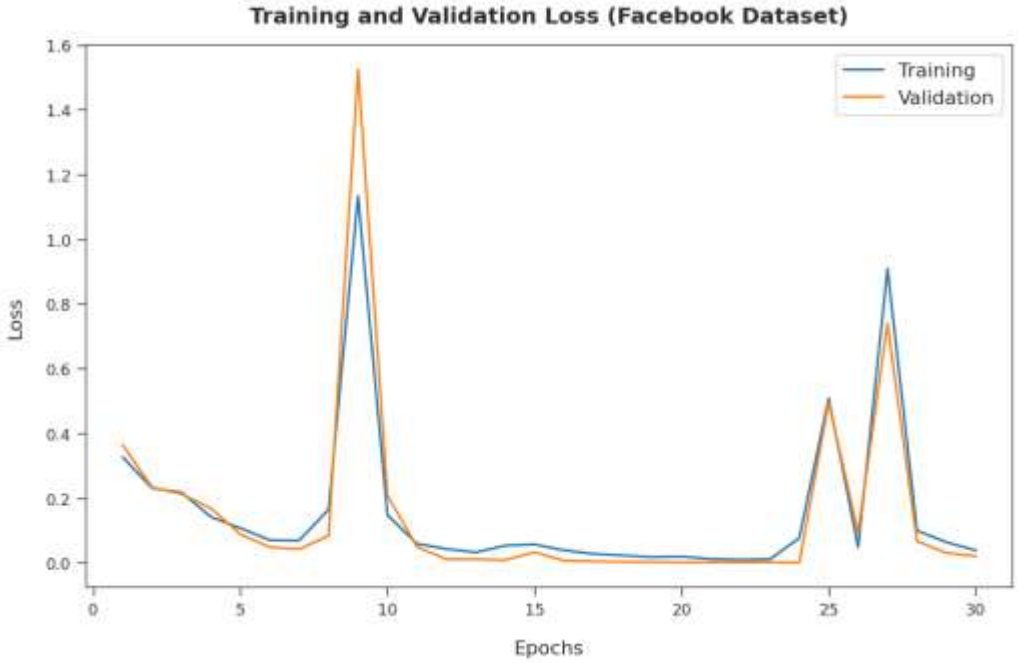

Figure 4: Loss analysis of DFA-MSVM technique on Facebook dataset

Fig. 3 illustrates the training and validation accuracy inspection of the DFA-MSVM model on Facebook dataset. The figure conveyed that the DFA-MSVM model has offered maximum training/validation accuracy on classification process.

Then, Fig. 4 exemplifies the training and validation loss inspection of the DFA-MSVM model on Facebook dataset. The figure reported that the DFA-MSVM model has offered reduced training/accuracy loss on the classification process of test data.

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

Table 1 demonstrates a comparative botnet classification outcomes of the DFA-MSVM model interms of TP, TN, FP, and FN on Facebook dataset.

Table 1: Comparative analysis of DFA-MSVM technique with existing approaches on Facebook dataset

Methods	True Positive	True Negative	False Positive	False Negative
DFA-MSVM	98.51	98.42	1.58	1.49
Naïve Bayes	97.29	97.94	2.06	2.71
Decision Tree	97.38	98.16	1.84	2.62
SVM Model	97.57	97.21	2.79	2.43
Bagging	97.25	97.88	2.12	2.75
Boosting	98.11	97.75	2.25	1.89

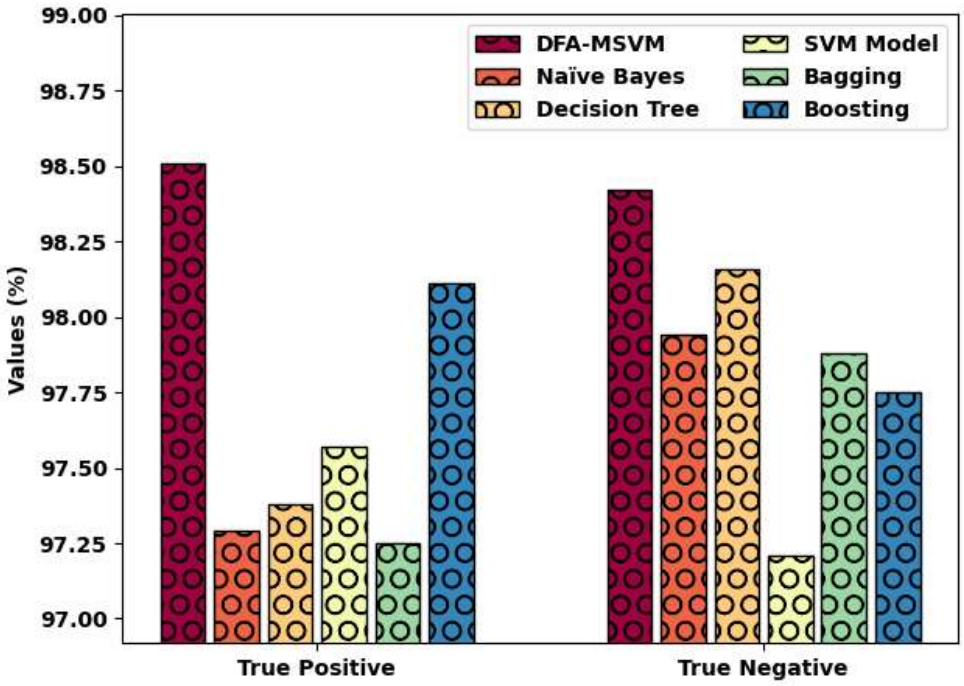

Figure 5: TP and TN analysis of DFA-MSVM technique on Facebook dataset

Fig. 5 indicates a brief comparative study of the DFA-MSVM model with existing models on Facebook dataset. The figure indicated that the bagging model has resulted in ineffective outcomes with TP and TN of 97.25% and 97.88% respectively. Along with that, the SVM model has gained slightly enhanced performance with the TP and TN of 97.57% and 97.21% respectively. In line with, the NB model has accomplished moderately closer TP and TN of 97.29% and 97.94% respectively. Followed by, the boosting and decision tree models have accomplished reasonable TP and TN values. Finally, the DFA-MSVM model has resulted in maximum TP and TN values of 98.51% and 98.42%.

Fig. 6 designates a brief comparison study of the DFA-MSVM model with existing models on Facebook dataset. The figure indicated that the bagging model has resulted to ineffective outcome with of 2.12% and 2.75% correspondingly. Likewise, the SVM approach has gained slightly enhanced performance with the FP and FN of 2.79% and 2.43% correspondingly. Similarly, the NB model has accomplished moderately closer FP and FN of 2.06% and 2.71% respectively. Besides, the boosting

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

and decision tree models have accomplished reasonable FP and FN values. Finally, the DFA-MSVM model has resulted in maximum FP and FN values of 1.58% and 1.49%.

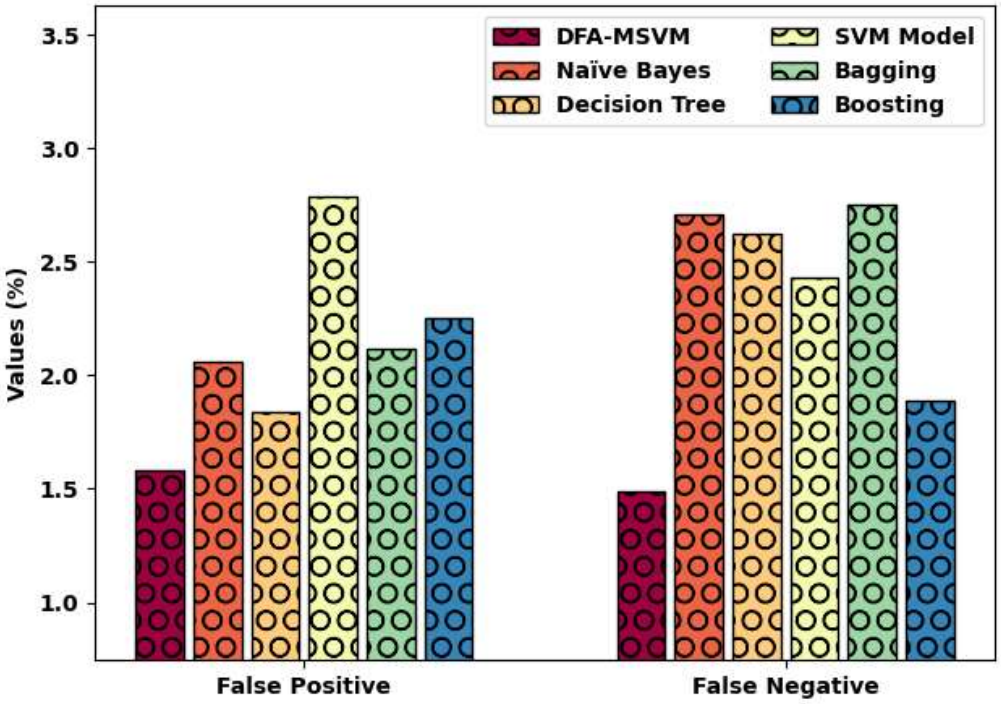

Figure 6: FP and FN analysis of DFA-MSVM technique on Facebook dataset

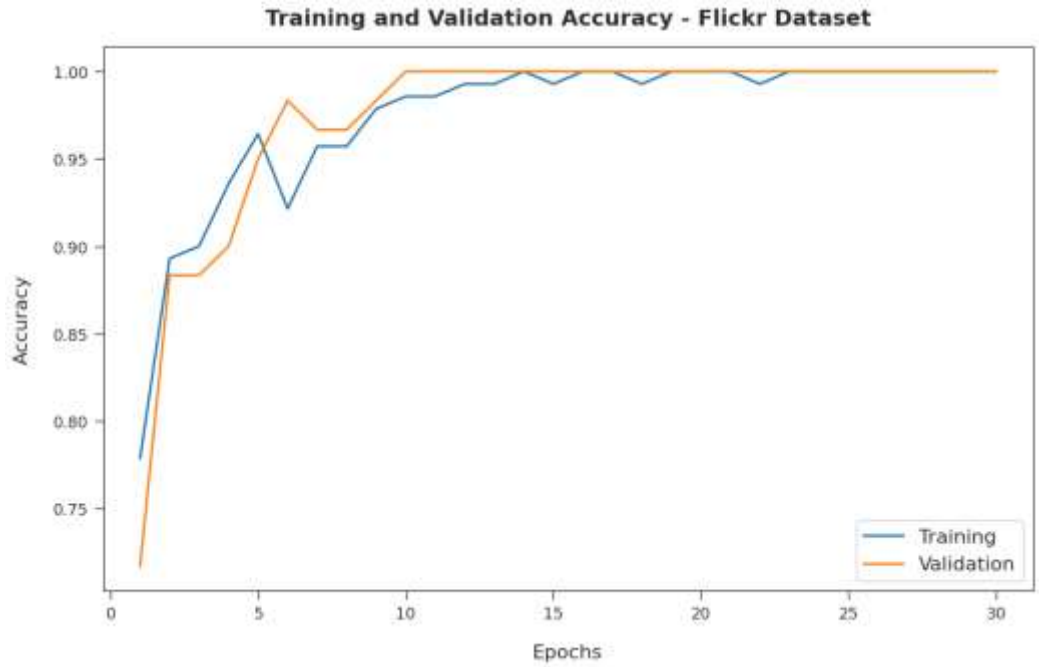

Figure 7: Accuracy analysis of DFA-MSVM technique on Flickr dataset

Fig. 7 shows the training and validation accuracy inspection of the DFA-MSVM model on Flickr dataset. The figure conveyed that the DFA-MSVM model has offered maximum training/validation accuracy on classification process.

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

Next, Fig. 8 demonstrates the training and validation loss inspection of the DFA-MSVM model on Flickr dataset. The figure described that the DFA-MSVM approach has offered reduced training/accuracy loss on the classification process of test data.

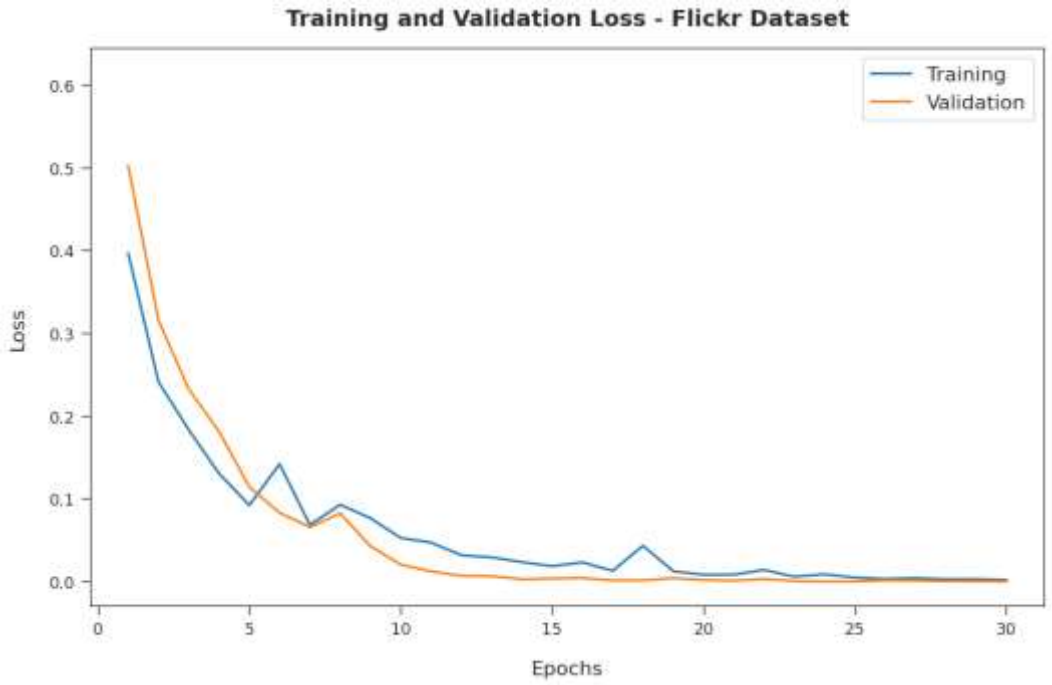

Figure 8: Loss analysis of DFA-MSVM technique on Flickr dataset

Table 2 demonstrates a comparative botnet classification outcomes of the DFA-MSVM model interms of TP, TN, FP, and FN on Flickr dataset [20].

Table 2: Comparative analysis of DFA-MSVM technique with existing approaches on Flickr dataset

Methods	True Positive	True Negative	False Positive	False Negative
DFA-MSVM	99.06	98.95	1.05	0.94
Naïve Bayes	97.47	98.04	1.96	2.53
Decision Tree	97.96	96.99	3.01	2.04
SVM Model	96.58	98.13	1.87	3.42
Bagging	98.12	97.24	2.76	1.88
Boosting	97.80	96.13	3.87	2.20

Fig. 9 depicts a brief comparative study of the DFA-MSVM model with existing models on Flickr dataset. The figure indicated that the bagging model has resulted in ineffective outcomes with TP and TN of 98.12% and 97.24% respectively. In addition, the SVM model has gained slightly enhanced performance with the TP and TN of 96.58% and 98.13% correspondingly. Similarly, the NB technique has accomplished moderately closer TP and TN of 97.47% and 97.04% correspondingly. Also, the boosting and decision tree methods have accomplished reasonable TP and TN values. Lastly, the DFA-MSVM model has resulted in maximal TP and TN values of 99.06% and 98.95%.

Fig. 10 specifies a brief comparative study of the DFA-MSVM model with existing models on Flickr dataset. The figure indicated that the bagging model has resulted in ineffective outcomes with of 2.76% and 1.88% respectively. Also, the SVM model has gained slightly enhanced performance with the FP and FN of 1.87% and 3.42% respectively. Besides, the NB model has accomplished moderately closer FP and FN of 1.96% and 2.53% correspondingly. Followed by, the boosting and decision tree models

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

have accomplished reasonable FP and FN values. Eventually, the DFA-MSVM model has resulted to maximum FP and FN values of 1.05% and 0.94%.

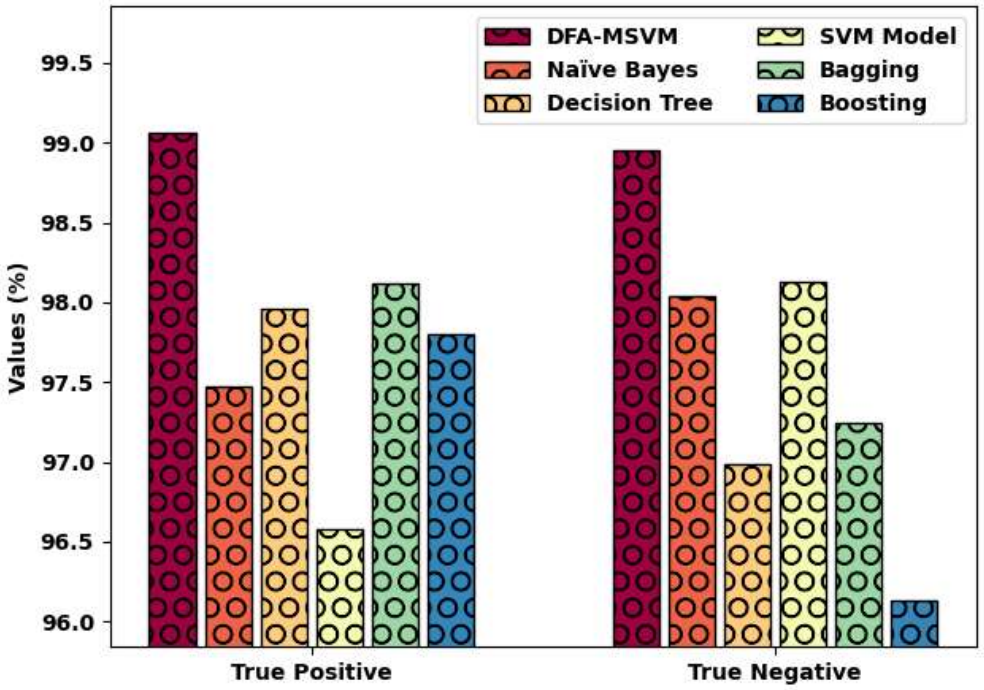

Figure 9: TP and TN analysis of DFA-MSVM technique on Flickr dataset

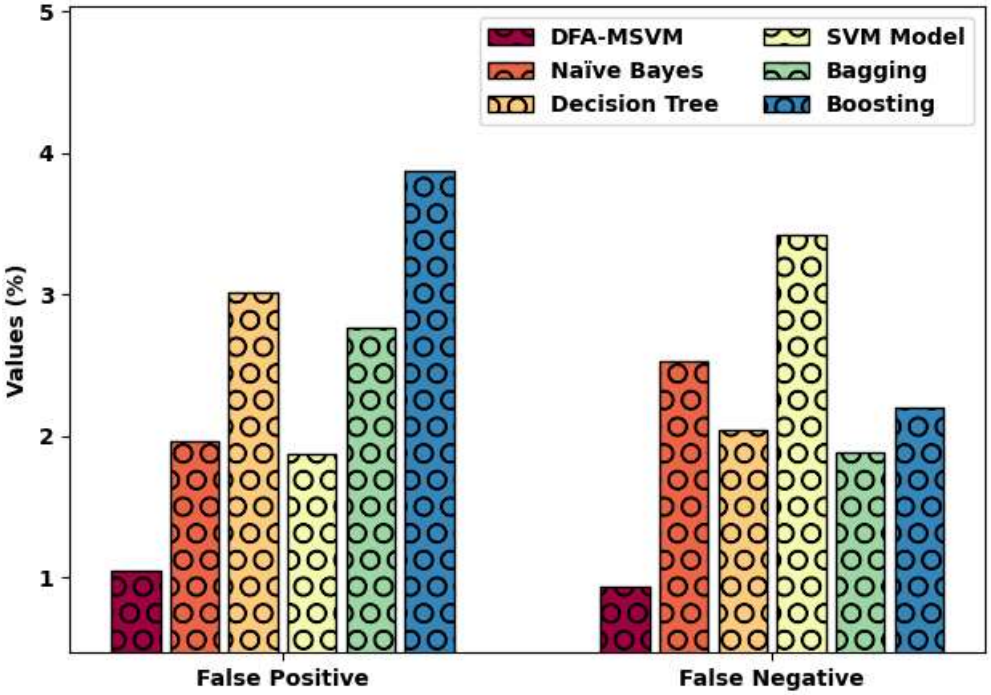

Figure 10: FP and FN analysis of DFA-MSVM technique on Flickr dataset

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

4. Conclusion

In this article, a novel DFA-MSVM model has been developed for recognition and classification of botnet for information security. For effectual recognition of botnets, the DFA-MSVM model involves data pre-processing at the initial stage. Besides, the MSVM model is utilized for the identification and classification of botnets that exist in the network. In order to optimally adjust the SVM parameters, the DFA is utilized and consequently resulting in enhanced outcomes. The presented DFA-MSVM model has the ability in accomplishing improved botnet detection performance. A wide-ranging experimental analysis is performed and the results are inspected under several aspects. The experimental results indicated the supreme outcomes of the DFA-MSVM model over existing methods.

References

[1] Alqatawna, J.F., Ala'M, A.Z., Hassonah, M.A. and Faris, H., 2021. Android botnet detection using machine learning models based on a comprehensive static analysis approach. Journal of Information Security and Applications, 58, p.102735.

[2] Khan, R.U., Zhang, X., Kumar, R., Sharif, A., Golilarz, N.A. and Alazab, M., 2019. An adaptive multi-layer botnet detection technique using machine learning classifiers. Applied Sciences, 9(11), p.2375.

[3] Pokhrel, S., Abbas, R. and Aryal, B., 2021. IoT Security: Botnet detection in IoT using Machine learning. arXiv preprint arXiv:2104.02231.

[4] Joshi, C., Bharti, V. and Ranjan, R.K., 2021. Botnet Detection Using Machine Learning Algorithms. In Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences (pp. 717-727). Springer, Singapore.

[5] Xing, Y., Shu, H., Zhao, H., Li, D. and Guo, L., 2021. Survey on botnet detection techniques: Classification, methods, and evaluation. Mathematical Problems in Engineering, 2021.

[6] Kumar, K., 2021. Comprehensive Method of Botnet Detection Using Machine Learning. International Journal of Open Source Software and Processes (IJOSSP), 12(4), pp.1-25.

[7] Lee, S., Abdullah, A., Jhanjhi, N.Z. and Kok, S.H., 2021. Honeypot Coupled Machine Learning Model for Botnet Detection and Classification in IoT Smart Factory–An Investigation. In MATEC Web of Conferences (Vol. 335, p. 04003). EDP Sciences.

[8] Hosseini, S., Nezhad, A.E. and Seilani, H., 2022. Botnet detection using negative selection algorithm, convolution neural network and classification methods. Evolving Systems, 13(1), pp.101-115.

[9] Joshi, C., Ranjan, R.K. and Bharti, V., 2021. A Fuzzy Logic based feature engineering approach for Botnet detection using ANN. Journal of King Saud University-Computer and Information Sciences.

[10] Tikekar, P.C., Sherekar, S.S. and Thakre, V.M., 2021, November. Features Representation of Botnet Detection Using Machine Learning Approaches. In 2021 International Conference on Computational Intelligence and Computing Applications (ICCICA) (pp. 1-5). IEEE.

[11] Alharbi, A. and Alsubhi, K., 2021. Botnet Detection Approach Using Graph-Based Machine Learning. IEEE Access, 9, pp.99166-99180.

[12] Popoola, S.I., Adebisi, B., Ande, R., Hammoudeh, M., Anoh, K. and Atayero, A.A., 2021. smote-drnn: A deep learning algorithm for botnet detection in the internet-of-things networks. Sensors, 21(9), p.2985.

[13] Ashraf, J., Keshk, M., Moustafa, N., Abdel-Basset, M., Khurshid, H., Bakhshi, A.D. and Mostafa, R.R., 2021. IoTBoT-IDS: A novel statistical learning-enabled botnet detection framework for protecting networks of smart cities. Sustainable Cities and Society, 72, p.103041.

[14] Ibrahim, W.N.H., Anuar, S., Selamat, A., Krejcar, O., Crespo, R.G., Herrera-Viedma, E. and Fujita, H., 2021. Multilayer framework for botnet detection using machine learning algorithms. IEEE Access, 9, pp.48753-48768.

[15] Muhammad, A., Asad, M. and Javed, A.R., 2020, October. Robust early stage botnet detection using machine learning. In 2020 International Conference on Cyber Warfare and Security (ICCWS) (pp. 1-6). IEEE.

[16] Janga, V. and Edara, S.R., 2021. Epilepsy and Seizure Detection Using JLTM Based ICFFA and Multiclass SVM Classifier. Traitement du Signal, 38(3).

[17] Mehmood, Z. and Asghar, S., 2021. Customizing SVM as a base learner with AdaBoost ensemble to learn from multi-class problems: A hybrid approach AdaBoost-MSVM. Knowledge-Based Systems, 217, p.106845.

[18] Mirjalili, S., 2016. Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural computing and applications, 27(4), pp.1053-1073.

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022

[19] Meraihi, Y., Ramdane-Cherif, A., Acheli, D. and Mahseur, M., 2020. Dragonfly algorithm: a comprehensive review and applications. Neural Computing and Applications, 32(21), pp.16625-16646.
[20] Elngar, A. and KRIT, S.D., 2019. Performance Analysis of Machine Learning based Botnet Detection and Classification Models for Information Security. J. Cybersecur. Inf. Manag, pp.44-53.

DOI: https://doi.org/10.54216/JCIM.090106
Received: February 10, 2022 Accepted: June 22, 2022