

Strategic-level Framework with Combined Feature Engineering, Bio-inspired Optimization and ML for Detecting DDos Flooding Attacks in Cloud

Kalaivani M¹ and Padmavathi G²

¹⁻²Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women
Coimbatore, India

Email: kalaivanim@gmail.com, padmavathi.avinashilingam@gmail.com

Abstract—Business and public enterprises have transitioned to cloud-based services. Most of the service-based companies do their core business activities using the cloud services allowing more accessibility, allowing multiple users to share the data regardless of their location. These new technologies also attract a lot of intruders and they use DDos as one of the main methods of attack. With increased amount of data and services being on cloud services Early detection of such attacks are dominant to any business. New innovation also attracts a rampant increase in the volume and frequency of DDos attacks. New generation mechanism and technique are required for deal with new types of DDos attacks. With Increasing number of businesses, educational institutions, cultural organizations relying on cloud services, the detection system should not only be effective but also affordable to be masses. Thus, in this work, a proposed strategic-level framework that combines feature engineering mechanisms with Bio-Inspired optimization technique and machine learning. The models also are verified with opposite confirmation and tested for area-under-curve analysis. It provides a wide range of reliable solutions to avoid data overload problems while detecting direct DDos flood attacks on Cloud. The framework is tested on the new DDos test database (CICDDoS2019) with the DT classifier. To validate the proposed system, The other machine learning techniques are selected from literature and compared the obtained results. Test results showed that the use of feature engineering techniques with the Dragonfly optimization algorithm and Decision Tree separator achieved a high accuracy rate (98.69%) in the detection of DDos flood attacks in a cloud environment. The results suggest that the proposed strategic level framework could achieve better results in DDos flood attacks detection on Cloud by promising reductions in cargo loads and times.

Index Terms— Cloud Computing, Security, DDos Flooding Attacks, Feature Engineering, Machine Learning.

I. INTRODUCTION

Cyber security refers to the use of security measures to provide privacy, integrity and data access. Cloud Computing is defined as the delivery of computer services including servers, storage, data, network, software, statistics, and intelligence — on the web (“cloud”) to deliver faster innovation, flexibility, and scale economy. Three forms of distributed computing administrations can be distinguished [1]: SaaS (software as a service),

PaaS (platform as a service), and SaaS (infrastructure as a service) (IaaS). However, it could very well be that way distributed as a private, public, public or mixed cloud. Distributed Denial of Service (DDoS) attacks are one of the major challenges and disruptions to computer computing that affect access to services. [2, 3] DDoS assaults in a cloud environment are not quite the same as DDoS assaults on conventional enterprises. In addition to the effects of DDoS attacks such as service disruption, loss of revenue generated downtime, negative impact on product image, cost of reducing attacks, etc., Additionally, there are unforeseen effects within the cloud, such as additional financial charges brought on by programmed estimates and costs. of extra energy used, unintentional backlash to distributed computing devices, information transfer and management starting in one cloud region then moving onto another, and afterwards negative effects brought on by shared VMs. Cloud DDoS is essentially an attack on Economic Denial of Sustainability (EDoS). [1].

The DDoS attacker plans to compromise and take a large number of weak animals, known as zombies, to aid in or launch a planned attack against the target. Burglary has been identified as one of the primary targets of such investigations, which have continued to grow in scope and complexity [4]. A web-based security problem known as denial-of-service (DoS) occurs when a targeted attack on a client renders it inaccessible or renders useless the assistance provided to actual customers. The complexity, durability, expense, and specific seller characteristics of modern business equipment and arrangements make it simple to test and execute DDoS systems in a cloud environment. Networks struggle to distinguish between beneficial and harmful streams. Although generally new, the use of AI computations to identify DDoS has so far shown promise [5-8, 10]. They work well when equipped with thorough knowledge. The quality, quantity, and metrics of information preparation have a significant impact on the viability of Mechanical Learning models. According to reports, the majority of analyses have relied on informational databases like KDD 99, NSL-KDD, DARPA and CAIDA, etc., which do not include recent DDoS attacks. The primary objectives of this study include,

(1) a proposed critical level structure that comprehensively coordinates machine learning and highlight design efforts to handle differentiating DDoS attacks; (2) system planning that takes into account turn of events and approval of machine learning models; and (3) extremely advanced security. DDoS direct attack rates on distributed computing with the proposed structure.

The structure of the full document is as follows: A capacity related to Section 2's investigation is provided. The vital structure and exploring strategy are examined in Section 3. Designing and security techniques are examined in Section 4. Development and analysis are presented in Section 5, and a specific execution evaluation and execution assessment are shown in Section 6. In the final section, there were some concluding remarks and a discussion of future work.

II. RELATED WORK

At this stage, the literature on DDoS attack detection methods is summarized and discussed. The recent rise in DDoS attacks has attracted a great deal of interest from researchers as they avoid traditional network-based detection methods. In the first three months of 2018, DDoS assaults were launched against targets in 79 and 84 countries, respectively, and a bigger portion of 95% of those attacks took place in the top 10 countries [12]. Alkasassbeh et al. [12] make use of cutting-edge DDoS attack data obtained from multiple network stages, such as HTTP flood and SIDDOS. The work revolves around a close assessment in which 27 factors are taken into consideration, and uses machine learning methods including the Multilayer Perceptron, the Nave Bayes, and the Arbitrary Backwoods to ascertain precision.

A Support Vector Machine (SVM) screening tool was suggested by Liao et al. [13]. The similarity of the bot's capacity to access pages foreshadows revelation. In order to identify comparable cases, they used a mood matching computation in conjunction with a similar application recurrence succession element to store clients' application samples. An independent convention for recognizing DDoS attacks motivated by thought is provided by Gao et al. [14]. It has been determined that five factors can be used for machine based DRDoS getting (disavowal of a distributed show of administration) in an autonomous manner. Highlights include the number of parcels (per unit time other than TCP or UDP header), bundle size shipped off the objective, the absolute number of bundles shipped off the objective, comparison of parcel numbers sent back from focus at the unit time, and the maximum number of bundles sent per unit time. [15] uses the Naive Bayes AI approach with highlight choice. Using a Bayes-driven region and eight speculative elements from the CAIDA'07 data base, a DDoS attack is identified by mechanical picking up. To screen for potential assaults and make disclosures, numerical equations for the Naive Bayes technique were gradually changed. 16 elements from the CAIDA'07 dataset are the most popular option in [15]. It is suggested that a coordinated approach be used to choose a calculation in which important metrics such as data gain, chi-square, gain proportion, SVM, proclivity level, reliefF, and the balance

of vulnerability in the scale are seen so that single focuses are taken into account as the all-outout limit. component to empower or debilitate admission to the element in the waitlist. The use of entropy and granular computerization in highlight determination is presented in the work by Khan et al. [16] for the focusing of DoS attacks. Entropy is calculated for each of the seven examinations included in the NSL-KDD'09 data set with a total of fifty cases, and a weight is assigned to each component taking the confounding estimate into account. In [17], using two informative sets, ISOT and ISCX, the collapsing method1 is used to select different combinations of attributes by self-centered search using a hereditary calculation, and then test those components using the C4.5 calculation. Highlights that provide a higher identification rate with C4.5 are the ones that are highlighted.

After selecting the TCP/IP subject data component, several AI models focus on the reenacted flood SYN flood discovery in the cloud climate [18]. After pre-handling and element removal, a cross-sectional interaction is familiar with choosing a component that provides the proper list of capabilities after collecting the typical highlights provided by measures from ReliefF, data gain (IG), and gain percentage (GR). [19] investigate the use of AI computations to identify DDoS attacks in a cloud environment. In this study, a simple 5 set is dissected using a C4.5 choice tree calculation that takes the addition proportion into account. For improved outcomes, the procurement module is also compatible with signature-based acquisitions. The untried Bayes and K-implies techniques are the other two that are tested for evaluation.

An IDS that works beautifully for your company with both HIDS and NIDS of a cloud kind has been proposed by P. Ghosh and A. Kumar in [20]. They used the data gain technique to select the pieces that would work best for improving the recognition accuracy. The neural network was used to distinguish between the different types of attacks, and the NSL-KDD knowledge base was used to pinpoint the location framework. The organisation association stage was carried out using the K-closest neighbour technique. Regarding the report on the split's effects, the cloud regulator was informed. In light of Discrete Deferential Evolution (DDE) and Decision Tree, Popoola, Ebenezer, and A.O. Adewumi recommended a viable network access determination cycle component in [21]. (C4.5). Seven AI calculations in the NSL-KDD data set were employed by the Weka device, and they were all approved. DDE was used to reduce the amount of the dataset, and the characterization findings obtained demonstrated significant improvements in the obtaining precision as well as a reduction in the preparation and testing times.

A combination of classifier models that supported using tree-based calculations to determine network entrance were proposed by Kevric, Jasmin, and S. Jukic in [22]. The NSL-KDD informational index was used to evaluate the framework. The findings indicate that combining class classifiers maintains absolute outline decide results that are unquestionably superior to using each separator. Because of the adjacent cloud organisation, S. Mehibs and S. Hashim advocated IDS in [23]. To identify inappropriate organisational behaviour, a fuzzy c mean calculation has been used. Their ability to recognise new attacks was the fundamental advantage of their assimilation technique. The suggested framework was evaluated using the KDD99 dataset, which has a high localization rate and low misleading positive caution. As an essential classifier, Ngoc & T. Pham et al. suggested IDS in [24] using supporting and packing outfit procedures near to tree calculation. The suggested framework was tested using the NSL-KDD informative index. The results showed that, depending on the segment precision and deception level, using the J48 classifier wiring model provides excellent performance when working with a base configuration of 35 chosen features. The cloud space's IDS has been raised by A. Antony and R. Priyadarshini in [25]. They dealt with pre-handling network data using a cuckoo enhancement-based method to increase identification accuracy. The results demonstrated that their IDS outperformed the competition.

III. PROPOSED STRATEGIC FRAMEWORK AND APPROACH

The most perfect identification in a specific DDoS dataset can be achieved by utilizing AI algorithms that are designed from the perspective of design and bio-inspired improvement techniques. In this assessment, a fundamental framework is suggested that includes a detailed aspect for executives and is followed by advancements in AI. emphasizes that the problem of overfitting should be solved through AI outcomes. Fig. 1 illustrates the proposed structure.

IV. FEATURE ENGINEERING

The data utilized as a research model in this paper is provided by the Canadian Institute for Cybersecurity2 et al [26]. The CICDDoS2019 data sets are a standalone label that is labeled and used worldwide by universities, the private sector, and independent researchers. CICDDoS2019 has a harmless attack and as a result of the latest

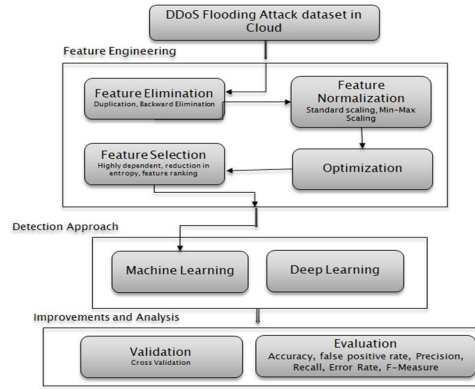


Fig 1. Proposed Strategic-level Framework for DDoS Flooding attack detection in Cloud

DDoS, which is similar to real-world data. Additionally, network traffic is decomposed using CICFlowMeter-V3 by mark stream, taking into account the time stamp, source and objective IP addresses, source ports, conventions, and attacks on CSV files. Approximately 121,980 (41.4%) records in the CICDDoS2019 data set were classified as "normal traffic," whereas 172,647 (58.6%) records were obtained as "attack traffic." The recently issued CICDDoS2019 informational index, which comprises a wide variety of DDoS attacks and pressure spaces for current informational indexes, is used to update the suggested model. The data collection includes the greatest variety of DDoS flood attempts that will be transmitted using TCP/UDP network and application layer protocols. The data base was gathered over the course of two distinct days for testing and preparation. The test information was arranged on March 11, 2019, and it includes 7 DDoS attacks. The practice set was hung on January 12, 2019, and it contains 12 different types of DDoS attacks. Both PCAP records and supported stream configuration designs for CICDoS2019 are available on the Canadian Institute for Cybersecurity2 website.

In the CICDDoS2019 dataset, As shown in Fig 2 there were 41.4% (121,980 records) classified as normal traffic and 58.6% (172,647 records) categorized as attack traffic.

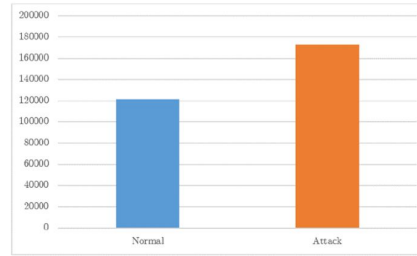


Fig 2. Bar chart showing the spreading of traffic type in the CICDDoS2019 dataset

A. Feature Elimination

RFE is a repetitive process in which features are measured based on a specific basic model of the Random Forest classification using the element value [27]. Pseudocode for RFE is in Algorithm 1

Algorithm 1: Pseudocode for the Recursive Feature Elimination (RFE) Algorithm

```

I/P : Training set X
Set of n features  $Y = \{f_1, f_2, \dots, f_n\}$ 
 $A(X, Y)$  # Ranking method
O/P: Final Ranks R
for  $j$  in  $1 : n$  do
    Rank set Y using  $A(X, Y)$ 
     $f^* \leftarrow$  last ranked feature in Y
     $R(n - j + 1) \leftarrow f^*$ 
     $Y \leftarrow Y - f^*$ 
end

```

RFE is a dull focus cycle wherein highlights are valued according to the value of their constituent parts. Method $A(X, Y)$ is used to evaluate highlights based on component value in Algorithm 1 above, where X addresses the preparation set and n number of highlights and Y addresses the first arrangement of components. The final degree of elements is covered by R . Every duplicate is evaluated for highlight value, and highlights that are only

loosely connected are removed. Following the discovery of the optimal number of elements, the appropriate subset set is selected in light of the above calculation's demonstration.

B. Feature Normalization

To put the mathematical data on the same scale, normalization is essentially applied. The huge number of digit component values in AI calculations should ideally not be more than the basic math values. The standardization approach often uses min-max scaling, when values between two provided numbers are being approximated. Using an estimate that falls between 0 and 1 is the method that is most commonly accepted (both consolidated). The min-max standardization formula is shown in figure (1) below. The standard worth is n_i of x_i , which is a collection of values ranging from $x_{\max}$, for example, the Maximum number, to $x_{\min}$, for example, the Minimum number. One of the most often used methods for estimating values between two given numbers is min-max scaling. Estimating between 0 and 1 is the method that is most frequently accepted (both consolidated).

$$n_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}$$

where n_i is normalized value of x_i is i th in a set of values from $x_{\max}$ (Maximum number) to $x_{\min}$ (Minimum number).

C. Dragonfly Optimization

The Dragonfly algorithm (DA) inspired by Static and Dynamic swarming behaviors of dragonfly in nature. It shows how it moves the individuals around the search space with respect to each other as well as food source and enemy [43]. Algorithm 2 is the pseudocode for DA.

Algorithm 2: Pseudocode for the Dragonfly Optimization Algorithm (DA)

```

Initialize the population  $X_i$  ( $i=1,2,\dots,n$ )
Initialize  $\Delta X_i$  ( $i=1,2,\dots,n$ )
while ( $t < \text{Max\_Iteration}$ )
    Evaluate each dragonfly
    Update (F) and (E)
    Update the main coefficients
    (i.e., w, s, a, c, f, and e)
    Calculate S, A, C, F, and E
    (using Eq (2) to (6))
    Update step vectors using Eq (7)
end while

```

There are three basic principles of swarming behavior: -

- Separation: It denotes the avoidance of individual static collision.
- Alignment: It denotes the velocity matching the individual to the neighborhood individual.
- Cohesion: It denotes the tendency of individuals to the neighborhood mass center.

Since survival is the main goal, therefore, all individuals are disturbed by external enemies and attracted to food sources. These parameters are statistically defined.

Separation: This is calculated using below equation (2):

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

Alignment: This represents the mean of the velocities of all neighbors using below equation (3):

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

Cohesion: this is calculated using the below equation (4):

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

Attraction towards a food source is calculated using below equation (5):

$$F_i = Y^+ - Y$$

Distraction outwards an enemy is calculated using below equation (6):

$$E_i = Y^- - Y$$

The behavior of the dragonfly is a combination of these five shown in figures (7).

$$\Delta Y_{t+1} = (a A_i + s S_i + c C_i + e E_i + f F_i) + w \Delta Y_t$$

When A = alignment weight, C = cohesion weight, e = enemy weight, f = food weight, s = separation weight, w = inertia weight. t = multiplication counter. Number of Iteration = 2000. Number of Dragonfly = 40. d = the calculated dimension of vector positions that charge flight step.

D. Detection Approach

To identify AI models, one uses data sets prepared for AI tests in a previous stage. It has been demonstrated that using AI in conjunction with highlight design can be an effective way to identify DDoS attacks. The area under the bend (AUC) is displayed at the exhibition evaluation stage to show the best results. It examines how the AI calculation behaves and analyses how effectively the knowledge base comprising key highlights performs magnificently for the use of different AI calculations. With the Decision Tree classifier, the structure is tested in this review using the most recent DDoS test data set (CICDoS2019). The other AI techniques are selected from writing and considered in light of the obtained findings in order to approve the suggested framework. A prominently used machine learning calculation is the choice tree calculation, also known as the C4.5/J48 computation [19]. This is seen in Fig. 3.

Algorithm 3: Pseudocode for the Decision Tree Algorithm (DT)

```

Create a root node N;
IF (T belongs to same category C)
{leaf node = N;
  Mark N as class C;
  Return N;
}
For i=1 to n
{Calculate Information_gain (A);}
ta = testing attribute;
N.ta = attribute having highest information_gain;
If(N.ta == continuous)
{find threshold;}
For(Each T in splitting of T)
if(T is empty)
{child of N is a leaf node;}
else
{child of N = dtree T}
calculate classification error rate of node N;
return N;

```

Fig 3. The pseudo code for the Decision Tree Classifier algorithm

Consistent and segmental highlights are both taken into account by the calculation. The calculation uses a high level approach for pruning trees to lessen the chance of disastrous characterization due to the high commotion or an exceptionally nitty-gritty preparation informative index. The components of the data set that will be further partitioned are contained in this copy dynamic calculation, which employs the eager strategy for division and triumph. Isolation can be a crucial execution boundary in any decision tree. Because the amount of cases isn't categorized as planned, the partition error is frequently identified by this. While creating a decision tree, the calculation appears to include continuous and stage-based highlights. This decision tree can be built in a vertical or even manner. The choice tree consists of different choice locations and leaf hubs, where the choice hubs determine the trait test and the leaf hubs address class values [19].

V. IMPROVEMENT AND ANALYSIS

An important part of the proposed strategic framework is the development of machine learning and analysis used to verify results and to avoid potential inaccuracies and deceptive accuracy. Details of the steps within the development of machine learning and analysis module are provided below:

A. Validation

The results of machine learning depending on the accuracy and level of errors in specific training formats and test data sets may invite overload problem. It frequently happens that the preparation data set is set up so that all points of view reflect similarly to those in the test data set. In such a situation, the overuse of preparing information may lead the exactness of the reading to be unreasonably high. This may give the mistaken impression that the model is being executed. K-crease cross confirmation is widely used to avoid this congestion issue and relies on the accuracy-related outcomes of AI. In the affirming wrap-up, k times' worth of preparing and evaluating is done randomly. The k different portions of the entire dataset are framed in each round. Information is separated into "k 1" portions for preparation and one split for testing. The classifier unwavering quality improvement used the segregated k-overlap cross-approval. randomly dividing the entire dataset into k "folds." Create a model on the dataset's folds k - 1 for every k-overlay. The model is then used to examine the

potential for kth overlap at that point. Record the error against each expectation. Until each k overlap has been removed as a testing set, these means are repeated. The cross-approval mistake, which is the average of k documented errors, will serve as the model's execution metric. The 10-crease cross-approval technique was used to produce a reliable exhibition of the classifier. Once that is done, the cross-approval gauge of the MISE forecast error is calculated using the condition (8).

$$\widehat{MISE}_{CV} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i^{k(i)})^2 = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{f}(x_i, \alpha^{k(i)}))^2$$

For each round accuracy score is measured and the test split is modified. ensures that the boundaries applied don't result in over foundation in the case that the precision adjustment remains within a narrow range of values.

B. Evaluation

The test stage is the final part of the suggested framework, and it is here that various exactness focuses and readings from other selected measurements can be compared to determine the optimum way to set the AI model for the greatest results. By reorganizing gaining the data according to just one categorization in all information circumstances, testing can avoid the exactness chaos that leads to a false sense of model precision. It offers no benefits of both positive and negative deceptive reading, yet it can still demonstrate high levels of accuracy. The recipient's presentation factor diagram (ROC) is set up between the positive and negative readings, with the lower bend (AUC) providing real precision to the model characterization, to avoid this uncertainty.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

Making execution measures is a key step in comprehending how a model exhibits. In this study, the recently delivered CICDDoS2019 dataset [20] is used to evaluate the suggested model. The data set includes a wide range of DDoS flood attacks that can be carried out using various TCP/UDP-based frameworks. On two different testing days, the preparation and test data sets are collected. The practice set contains 12 different DDoS attacks that were recorded on January 12, 2019. The planning day included a variety of attacks, including SYN, SNMP, TFTP, UDP, LDAP, NetBIOS, WebDDoS, DNS, NTP, MSSQL, UDP-Lag, and SSDP DDoS. SYN, MSSQL, UDP-Lag, LDAP, UDP, PortScan, and NetBIOS are the seven DDoS attacks that were launched on March 11, 2019, the eleventh day of testing. Utilizing CICFlowMeter devices, more than 801 stream attributes from the data set were eliminated [24]. On the website of the Canadian Institute, you can view the data set under PCAP record and streak. Different metrics were developed for this review. In terms of classification accuracy, exactness, review, mistake rate, F-Score, and 10-crease invert confirmation, execution testing is validated. These metrics are determined using the terminology mentioned above [22], and they are frequently used in mediation frameworks.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad [9]$$

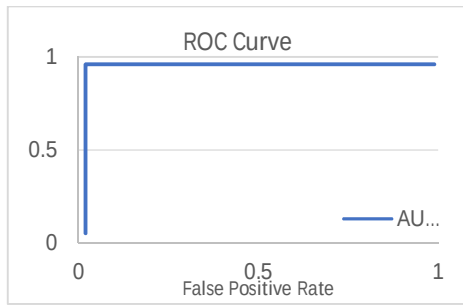
$$Precision = \frac{TP}{TP + FP} \quad [10]$$

$$Recall = \frac{TP}{TP + FN} \quad [11]$$

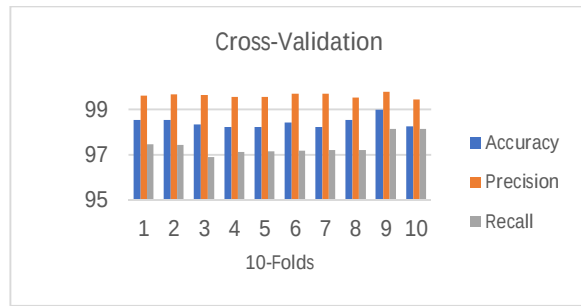
$$F-Score = \frac{2 \times (Recall \times Precision)}{Recall + Precision} \quad [12]$$

Fundamentally, the Receive Performance Curve (ROC) is employed to measure model execution exactly. The ROC bend illustrates the connection between the real and fake class boundaries and the x-hub structure, with the incorrect x-pivot y having a positive value of the various benefits of the various qualities between 0.0 and 1.0 [26]. The difference between the false positive rate and the true positive level is estimated by the area under the ROC Curve. Figure 4 depicts the suggested visual system's ROC bend, which displays an AUC of 98.8, meaning that our model can correctly identify 98.8% of positive and negative classes.

True Positive (TP) & True Negative (TN) represent positively predicted values in the proposed model. By contrast the classification is illustrated by False Positives (FP) and False Negatives (FN). For this validation of



ROC Curve of proposed detection framework



Bar model shows 10-fold validation process of DDoS Flooding attack detection model

the proposed model, the value of k is 10 to wrap the data. Here the database is divided into 10 groups. In all validations the first group is used as a test database and the remaining groups are used as a training database. At the time when the model was created from test data, the model should be tested with the remaining groups. Once the initial test is complete it will be discarded. Then, the following verification is continued. When the 10-fold verification process is completed, the total model rating must be calculated as shown in the figure 5.

The proposed model is compared with various old ML techniques from the literature. In this paper, there are six different algorithms that have been compared based on their performance, NB, Decision Tree (DT), Random Forest (RF), SVM, Booster and Logistic Regression (LR). Table I represents the classification metrics for the proposed model with different alternatives. The results show that the proposed level level model works better with accuracy, memory, and F-measure of 0.99, 0.98, 0.98 and 98.89 accuracy than other ML algorithms. It has been noted that the proposed model performed very well, followed by LR and SVM.

TABLE I. TRIAL OF THE PROPOSED MODEL'S EXECUTION AND OTHER ILLUSTRATIVE ML CALCULATIONS. COMPARING OUR SUGGESTED TECHNIQUE TO OTHER ESTIMATION CALCULATIONS, IT PERFORMS EXCELLENTLY

Techniques	Precision	Recall	F-Score	Accuracy
<i>NB</i>	1	0.17	0.29	57
<i>DT</i>	0.7	0.99	0.82	77
<i>Booster</i>	0.76	0.99	0.86	84
<i>RF</i>	1	0.74	0.85	86
<i>SVM</i>	0.99	0.88	0.93	93
<i>LR</i>	0.93	0.99	0.96	95
<i>Proposed DT-DOA</i>	<i>0.99</i>	<i>0.98</i>	<i>0.98</i>	<i>98.89</i>

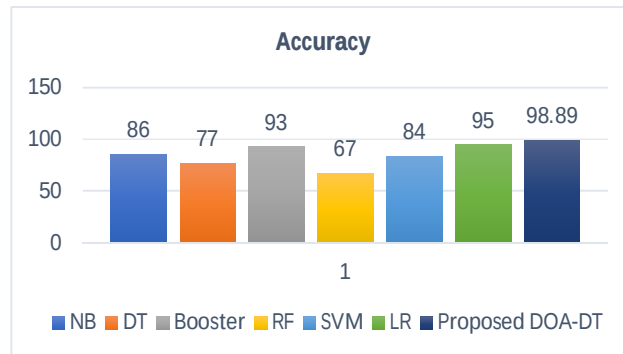


Fig 6. Accuracy of the proposed Strategic level framework

Figure 12 shows accuracy levels of proposed models trained with the CICDDoS2019 database [26] and Figure 5-7 shows some metrics performance. The proposed strategic element-based strategy with bio-inspired development strategies and machine learning demonstrates excellent performance, achieving 98.89% accuracy, 99% accuracy and 99% memory. DDoS Flood Attack against the Cloud network has ensured that machine learning combined with feature engineering and BI development techniques are highly efficient for the detection of DDoS flood attacks.

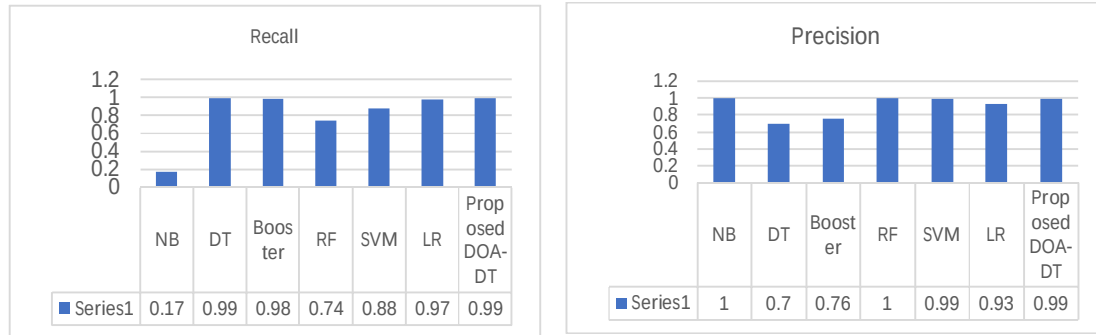


Fig 7. Recall and Precision of the proposed method

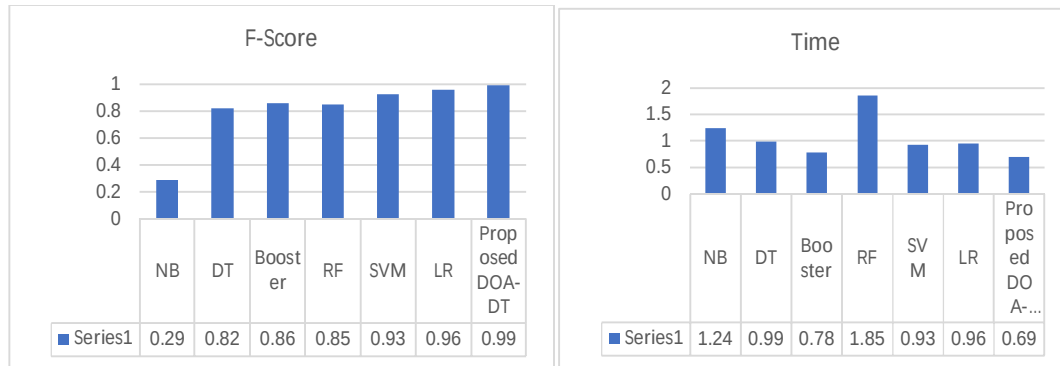


Fig. 8.F-Score and Time of the proposed Strategic level framework

When compared to other well-known ML algorithms, evaluation of this suggested model has revealed that it provides the highest test metrics in terms of memory, exactness, F focuses, and precision. Using a 10-fold confirmation approach, the separator's dependable functioning was achieved. This suggested program's effects are contrasted with the writing. The results obtained demonstrate that the proposed strategy's demonstration is quite persuasive when compared to other results obtained and appears to be very encouraging in the area of DDoS Flooding attacks on Cloud.

VII. CONCLUSION AND FUTURE WORK

The DDoS flood attack, which is easing up in the cloud region, is thought to be one of the most powerful recent attacks, seriously damaging the entire network structure. In this research, a model is put out for a new fundamental level structure that combines bio-inspired architecture with artificial intelligence to detect DDoS flood attacks on cloud environments. The recently delivered CICDDoS2019 data base was used to create and evaluate our suggested model. The data base includes a complete list of attacks and ongoing DDoS floods. Comparing our model to other known ML techniques, evaluation reveals that it performs extraordinarily well in terms of memory, accuracy, F focus, and precision. In comparison to existing feature choosing techniques and characterization strategies, the proposed Strategic level system with designing and Bio-Inspired improvement procedures using AI methods provides greater class precision and better learning time. With this proposed methodology, testing on various data sets should eventually be achievable. A two-part system has been used in the ongoing endeavor to separate the information stream into legitimate and malicious structures. In any case, it is crucial to clearly distinguish between each type of assault. To expand this effort into a multi-stage structure is

what it signifies. Additionally, it should be feasible to recreate a cloud-based network with numerous locations and populations in order to create a different data set that effectively manages current online traffic.

REFERENCES

- [1] Aanshi Bhardwaj, Veenu Mangat, Renu Vig, Subir Halder, Mauro Conti, Distributed denial of service attacks in cloud: State-of-the-art of scientific and commercial solutions, *Computer Science Review*, Elsevier, Volume 39, 2021, 100332, ISSN 1574-0137
- [2] Casey Crane, The 15 top DDoS statistics you should know in 2020, 2019,(accessed Nov 14, 2019). URL <https://cybersecurityventures.com/the-15-top-ddos-statistics-you-should-know-in-2020>.
- [3] Tomer Shani, Updated: This ddos attack unleashed the most packets per second ever. here's why that's important, 2019, (Accessed September 3, 2019). URL <https://www.imperva.com/blog/>.
- [4] Jay Thakkar, DDoS attack statistics: A look at the most recent and largest DDoS attacks, 2019, (Accessed Oct 19, 2020). URL <https://sectigostore.com/blog/ddos-attack-statistics-a-look-at-the-mostrecent- and-largest-ddos-attacks/>.
- [5] DDoS attacks Trend Report, 2018. https://www.cdnetworks.com/CDNetworks_Q3_2017_DDoS%20Attack%20Trends%20Report_EN_201712.pdf. Accessed February 26, 2018.
- [6] Azab, A., Alazab, M., and Aiash, M., Machine learning based Botnet identification traffic, 2016 IEEE Trustcom/BigDataSE/ISPA, Tianjin, 2016, pp. 1788–1794.
- [7] Agrawal, P.K., Gupta, B.B., and Jain, S., SVM based scheme for predicting number of zombies in a DDoS attack, 2011 European Intelligence and Security Informatics Conference, Athens, 2011, pp. 178–182.
- [8] Chonka, A., Zhou, W., Singh, J., and Xiang, Y., Detecting and tracing DDoS attacks by intelligent decision prototype, 2008 Sixth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom), Hong Kong, 2008, pp. 578–583.
- [9] Jalil, K.A., Kamarudin, M.H., and Masrek, M.N., Comparison of machine learning algorithms performance in detecting network intrusion, 2010 International Conference on Networking and Information Technology, Manila, 2010, pp. 221–226.
- [10] Mirjalili S (2016) Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Comput Appl* 27(4):1053– 1073
- [11] Xiao, Le, et al. "A protocol-free detection against cloud oriented reflection DoS attacks." *Soft Computing* 21.13 (2017): 3713-3721.
- [12] Alkasasbeh, Mouhammd, et al. "Detecting distributed denial of service attacks using data mining techniques." *International Journal of Advanced Computer Science and Applications* 7.1 (2016).
- [13] Q. Liao, H. Li, S. Kang, and C. Liu, "Application layer ddos attack detection using cluster with label based on sparse vector decomposition and rhythm matching," *Security and Communication Networks*, vol. 8, no. 17, pp. 3111–3120, 2015.
- [14] Gao, Y., Feng, Y., Kawamoto, J., Sakurai, K.: A machine learning based approach for detecting DRDoS attacks and its performance evaluation. In: 11th Asia Joint Conference on Information Security (AsiaJIS), pp. 80–86 (2016)
- [15] Singh, N.A., Singh, K.J.,De, T.:Distributed denial of service attack detection using Naive Bayes classifier through info gain feature selection. In: International Conference on Informatics and Analytics, p. 54 (2016)
- [16] Khan, S., Gani, A.,Wahab, A.W.A., Singh, P.K.: Feature selection of Denial-of-Service attacks using entropy and granular computing. *Arab. J. Sci. Eng.* 43(2), 499–508 (2018)
- [17] Singh, K.J.,De,T.:Efficient classification of DDoS attacks using an ensemble feature selection algorithm. *J. Intell. Syst* (2017). <https://doi.org/10.1515/jisys-2017-0472>
- [18] Al-Hawawreh, M.S.: SYN flood attack detection in cloud environment based on TCP/IP header statistical features. In: 8th International Conference on Information Technology (ICIT), pp. 236–243 (2017)
- [19] Zekri, M., El Kafhali, S., Aboutabit, N., Saadi, Y.: DDoS attack detection using machine learning techniques in cloud computing environments. In: 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech), pp. 1–7 (2017)
- [20] Partha Ghosh, A.K. Mandal and Rupesh Kumar, An Efficient Cloud Network Intrusion Detection System. *Information Systems Design and Intelligent Applications, Advances in Intelligent Systems and Computing*, Springer, 2015. vol. 339, pp. 91-99.
- [21] Popoola, Ebenezer, A.O Adewumi, Efficient Feature Selection Technique for Network Intrusion Detection System Using Discrete Differential Evolution and Decision. *International Journal of Network Security*, 2017. vol. 19, pp. 660-669.
- [22] Kevric, Dasmin, S. Dukic, A. Subasi, An effective combining classifier approaches using tree algorithms for network intrusion detection. *Neural Computing and Applications*, 2017. vol. 28, pp. 1051-1058.
- [23] S. Mehabs and S. Hashim, Proposed Network Intrusion Detection System Based on Fuzzy C Mean Algorithm in Cloud Computing Environment. *Journal of Babylon University/Pure and Applied Sciences*, 2017. vol. 26, pp. 27-35.
- [24] T. Pham, E. Foo, S. Suriadi and H. Jeffrey, Improving performance of intrusion detection system using ensemble methods and feature selection. *Australasian Computer Science Week Multi conference, ACM*, 2018. pp. 1-6.
- [25] A. Antony, R. Priyadharshini and D.L. Epiphany, Cuckoo Optimization based Intrusion Detection System for Cloud Computing. *International Journal of Computer Network and Information Security*, 2018. vol. 10, pp. 42-49.

- [26] Yuan, X., Li, C., Li, X.: DeepDefense: identifying DDoS attack via deep learning. In: International Conference on Smart Computing (SMARTCOMP), IEEE, pp. 1–8 (2017)
- [27] Lu, L., Feng, Y., Sakurai, K.: C&C session detection using random forest. In: 11th International Conference on Ubiquitous Information Management and Communication, p. 34 (2017)