ICT Express

Cost-Efficient DDoS Attack Detection: A Hybrid Filter-based Parameter Selection Approach

--Manuscript Draft--

Manuscript Number:	ICTE-D-21-00524
Article Type:	Research paper
Section/Category:	Area 1. AI for ICT Applications
Keywords:	computational efficiency; DDoS attack and detection; Machine Learning; parameter selection
Corresponding Author:	Dong-Seong Kim Kumoh National Institute of Technology School of Electronic Engineering Gumi, KOREA, REPUBLIC OF
First Author:	Gabriel Amaizu
Order of Authors:	Gabriel Amaizu
	Lucas Akpudo
	Dong-Seong Kim
Abstract:	Distributed denial of service (DDoS) attacks have become the most popular, frequent, and most devastating cyber attack accounting for a huge chunk of such attacks in recent times. In order to mitigate these rampant attacks, this study implements an ensemble network consisting of four differently configured deep neural networks (DNN) which is aimed at developing a robust and effective model that is free from high variance associated with traditional DNN models. Also, curse of dimensionality is mitigated by eliminating highly correlated features followed by a wrapperbased feature ranking by importance leading to a significant reduction in computational cost.
Suggested Reviewers:	Gaspard Gashema ggas06@yahoo.fr
Opposed Reviewers:	

Cost-Efficient DDoS Attack Detection: A Hybrid Filter-based Parameter Selection Approach

Amaizu Gabriel Chukwunonso, Ugochukwu Ejike Akpudo, Deong-Seong Kim*

ICT Convergene Research Center, Gumi, South Korea

Abstract

Distributed denial of service (DDoS) attacks have become the most popular, frequent, and most devastating cyber attack accounting for a huge chunk of such attacks in recent times. In order to mitigate these rampant attacks, this study implements an ensemble network consisting of four differently configured deep neural networks (DNN) which is aimed at developing a robust and effective model that is free from high variance associated with traditional DNN models. Also, curse of dimensionality is mitigated by eliminating highly correlated features followed by a wrapper-based feature ranking by importance leading to a significant reduction in computational cost.

Keywords: computational efficiency, DDoS attack and detection, machine learning, parameter selection.

1. Introduction

The role of information in our everyday lives and society cannot be overemphasized. The emergence and subsequent advancement in technology led to the Industrial Internet of Things (IIoT) which gave rise to technologies such as e-Health, smart homes, smart factories, and even autonomous vehicles. These advancements, sadly, have been followed by an increase in vulnerability of systems to cyberattacks [1, 2]. Distributed denial of service (DDoS) is regarded as the most sophisticated and frequent form of cyberattacks in recent memory[3]. In the occurrence of a DDoS attack, all network resources are exhausted by continuous and overwhelming requests thereby making it difficult or impossible for the network to respond to traffics from legitimate sources. This disruption is possible because the attacker uses a good number of compromised systems [4].

Since the beginning of 2020, a number of activities have mainly been conducted online as a result of the novel COVID-19 pandemic. This has subsequently translated to more attacks on online learning platforms, delivery agencies, and even medical research institutes, with Amazon AWS confirming that it mitigated the largest ever DDoS attack of 2.3 Tbps [5].

*Corresponding author

Email addresses: gabriel4amaizu@gmail.com (Amaizu Gabriel Chukwunonso), akpudougo@gmail.com (Ugochukwu Ejike Akpudo), dskim@kumoh.ac.kr (Deong-Seong Kim)

DDoS attacks can broadly be classified into two groups, bandwidth depletion attacks and resource depletion attacks. While the former floods the victims with illegitimate traffics, the later tries to exhaust all available resource of the victim [6]. In order to combat the ever present DDoS attack, the concept intrusion detecting systems (IDS) was coined. IDS over the the years have been broadly classified into three groups :- misuse-based, anomaly-based and hybrid IDS. Misuse based approaches tend to check network traffics for a known pattern of intrusion, anomaly-based methods identifies both known and unknown patterns of intrusion.

So far, there exist methods which have tried to tackle intrusion detection. Authors in [7] proposed a framework for software-defined networks. The scheme computes the cosine similarity of the packet-in rate for each entry port of switches in the SD-IoT and classifies traffic flow as attack flow or normal flow by using a computed threshold value for the cosine similarity. SVM is used together with Kernel PCA (KPCA) and genetic algorithm (GA) in [8]. KPCA redced the dimensions of the feature vectors while GA optimized various SVM parameters.

In [9], a privacy-preserving DDoS attack detection scheme is proposed by combining data encryption with perturbation encryption then an improved KNN algorithm was deplored in detecting DDoS attacks. Authors in [10] were able to obtain the hit rate from duration and count of entry flow and compute its gradient which

Cost-Efficient DDoS Attack Detection: A Hybrid Filter-based Parameter Selection Approach

Amaizu Gabriel Chukwunonso, Ugochukwu Ejike Akpudo, Deong-Seong Kim*

ICT Convergene Research Center, Gumi, South Korea

Abstract

Distributed denial of service (DDoS) attacks have become the most popular, frequent, and most devastating cyber attack accounting for a huge chunk of such attacks in recent times. In order to mitigate these rampant attacks, this study implements an ensemble network consisting of four differently configured deep neural networks (DNN) which is aimed at developing a robust and effective model that is free from high variance associated with traditional DNN models. Also, curse of dimensionality is mitigated by eliminating highly correlated features followed by a wrapper-based feature ranking by importance leading to a significant reduction in computational cost.

Keywords: computational efficiency, DDoS attack and detection, machine learning, parameter selection.

1. Introduction

The role of information in our everyday lives and society cannot be overemphasized. The emergence and subsequent advancement in technology led to the Industrial Internet of Things (IIoT) which gave rise to technologies such as e-Health, smart homes, smart factories, and even autonomous vehicles. These advancements, sadly, have been followed by an increase in vulnerability of systems to cyberattacks [1, 2]. Distributed denial of service (DDoS) is regarded as the most sophisticated and frequent form of cyberattacks in recent memory[3]. In the occurrence of a DDoS attack, all network resources are exhausted by continuous and overwhelming requests thereby making it difficult or impossible for the network to respond to traffics from legitimate sources. This disruption is possible because the attacker uses a good number of compromised systems [4].

Since the beginning of 2020, a number of activities have mainly been conducted online as a result of the novel COVID-19 pandemic. This has subsequently translated to more attacks on online learning platforms, delivery agencies, and even medical research institutes, with Amazon AWS confirming that it mitigated the largest ever DDoS attack of 2.3 Tbps [5].

*Corresponding author

Email addresses: gabriel4amaizu@gmail.com (Amaizu Gabriel Chukwunonso), akpudougo@gmail.com (Ugochukwu Ejike Akpudo), dskim@kumoh.ac.kr (Deong-Seong Kim)

DDoS attacks can broadly be classified into two groups, bandwidth depletion attacks and resource depletion attacks. While the former floods the victims with illegitimate traffics, the later tries to exhaust all available resource of the victim [6]. In order to combat the ever present DDoS attack, the concept intrusion detecting systems (IDS) was coined. IDS over the the years have been broadly classified into three groups :- misuse-based, anomaly-based and hybrid IDS. Misuse based approaches tend to check network traffics for a known pattern of intrusion, anomaly-based methods identifies both known and unknown patterns of intrusion.

So far, there exist methods which have tried to tackle intrusion detection. Authors in [7] proposed a framework for software-defined networks. The scheme computes the cosine similarity of the packet-in rate for each entry port of switches in the SD-IoT and classifies traffic flow as attack flow or normal flow by using a computed threshold value for the cosine similarity. SVM is used together with Kernel PCA (KPCA) and genetic algorithm (GA) in [8]. KPCA redced the dimensions of the feature vectors while GA optimized various SVM parameters.

In [9], a privacy-preserving DDoS attack detection scheme is proposed by combining data encryption with perturbation encryption then an improved KNN algorithm was deplored in detecting DDoS attacks. Authors in [10] were able to obtain the hit rate from duration and count of entry flow and compute its gradient which

is used in a backpropagation neural network for the detection of DDoS attacks. Whale Pearson algorithm is used in [11] for selecting 8 optimal features used for the intrusion detection system.

The importance of parameter selection for accurate data-driven DDoS attack detection cannot be overemphasized. It is necessary to ensure that only relevant parameters with high discriminative power are retained while redundant/irrelevant parameters are discarded. This ensures a reduced computational costs, eliminates the *curse of dimensionality*, over-fitting and classifier/predictor *confusion*. Several parameter selection techniques have been proposed in the past including filter-based and wrapper-based methods [12]. Wrapper methods have relatively high computational costs (especially on big data) while most filter methods are prone to several issues of instability [13] while being computationally efficient. In contrast, hybrid filter-based selection methods are more reliable as these methods combine more than a single statistical approach to ensure the input variable's characteristics are well assessed from multiple statistical standpoints thereby ensuring that the inter-relationships/dependence of the input variables is well assessed. From such statistical assessments, parameters can be selected based on discriminance levels/ranking while minimizing the curse of dimensionality, thereby minimizing computational costs. This the bedrock and first phase of this study.

In second phase we aim to improve the quality of the model in terms of accuracy and also reduce the high variance associated with neural networks. To achieve this, a meta classifier consisting of four distinctly configured neural networks is proposed. The DDoS attack detection methodology presented in this study is based on a stacked ensemble model intrusion classification scheme, a hybrid parameter selection procedure, and an oversampling methodology for solving data imbalance problems. The rest of this paper is organized as follows: Section 2 shows the system model of the proposed scheme. Section 3 presents the validation of the proposed methodology while Section 4 concludes the paper.

2. System Model

As discussed above, and can be seen in Fig. 1, this study is divided into two :- Phase 1 and Phase 2. In phase 1, the main aim is to obtain a list of optimal features from the given dataset which will then be used to train and evaluate the neural network in phase 2. This study has accounted for eleven DDoS attack types and

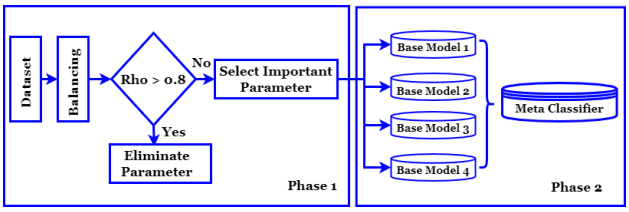


Figure 1. The proposed scheme showing the system flow and the proposed hybrid filter-based parameters selection

of course a benign traffic, hence phase 1 starts by ensuring that there are equal representation of all classes in the dataset to be used by performing data balancing using the synthetic minority oversampling technique (SMOTE). This becomes important as it eliminates the chances of arriving at a model skewed towards the majority class. Next a correlation test is performed to ascertain how one or more features are related to one another. For this test, a heuristic threshold ρ of 0.8 was set, meaning that features with correlation values above this threshold were eliminated. High correlation translates to high level of redundancy which translates to a higher cost.

On their own, correlation tests for parameter/feature selection are limited by instability issues, and their results, unreliable for industrial applications. By integrating another feature selection method like the Chi-squared for selecting important features, reliable DDoS detection accuracy can be ensured by using high-ranking parameters (most important parameters). Chi-square tests for the dependence between variables. With Chi-square, we aim to figure out those features that are highly dependent on the class and eliminate those features that are not dependent on the class. Features that are class independent contribute little or nothing to the model hence are not needed for classification. This processed is best explained in Algorithm 1.

$$\rho_{X,Y} = \frac{\mathrm{cov}(X,Y)}{\sigma_X \sigma_Y}, \tag{1}$$

Phase 1 takes as input the original featureset, denoted as F, it's dimension as D, a significance level P of 0.05 (which is a standard), and a heuristic correlation threshold R of 0.8. Now using Pearson's correlation given in Eqn 1 we calculate the correlation coefficients between all features in F while obtaining the p-value of each feature. If the obtained p-value is less than P and the correlation coefficient of both features is less than the threshold R, a new featureset F_s is returned with a dimension d which is less than the initial dimension D of F. Otherwise those features are ignored, as they are highly correlated. By the end of this operation, F_s,

Algorithm 1 Pseudo-code of proposed hybrid Parameter selection Method

Input : $F = f_1, f_2, f_3, \ldots f_k$: Original Featureset
D: Dimension of parameters
$P = 0.05$: Significance level
$R = 0.8$: correlation threshold

Output: F_s: Uncorrelated parameters
F_c: Optimal parameters

- Calculate the matrix C_{ij} of correlation coefficients between each f_i and f_j in F using (1)
- Calculate the p-value p_i for each feature

if $p_i \leq P$ *and* $C_{ij} \leq R$ **then**
  return F_s with dimension d ($d < D$)
else
  Ignore (F - F_s)
end
$chi\_array \leftarrow \varnothing$
max_no_of_features$\leftarrow$ 8
for *each feature in the set* F_s *(i = 1, ..., d)* **do**
  Calculate Chi-squared value (*chi_val*) between features in F_s using (2)
  $chi\_val \leftarrow chi.squared(F_{si}, F_{sj})$
  Append (i, chi_val) **To** *chi_array*
  Sort *chi_array* in descending order
end
$F_c = chi\_array[: max\_no\_of\_features]$
return F_c with dimension c ($c < d$)

which is a list of uncorrelated features is obtained.

Next an empty array (*chi_array*) is instantiated and using Eqn 2., the chi_squared value between each feature in F_s and the target class is computed. The values obtained is appended to the *chi_array* which is then sorted in descending order. F_c which is the optimal parameters is gotten by taking the first $max\_no\_of\_features$ (which is specified as 8) of *chi_array*. This will return F_c as the first 8 highest ranked features whose dimension is smaller than d and by extension smaller than D.

$$X_c^2 = \sum \frac{(O_i - E_i)^2}{E_i}. \quad (2)$$

where c is the degree of freedom, O is observed values and E_i is the expected values.

2.1. Proposed Meta Classifier

Phase two consists of a stacked ensemble model comprising of four base models (bm1, ...bm4) and a meta classifier as can be seen in Fig. 2. Each of the four base models is configured in a way that they are all different from each other. Each has an input layer, and output layer, a number of dense (hidden) layers, and one or more dropout layers within its architecture. One reason for this is to ensure that a wide range of dissimilar input is fed to the meta classifier.

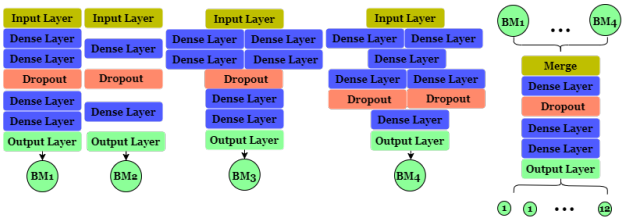


Figure 2. Phase 2 of the proposed scheme showing the configuration of the four base models and the meta classifier. Also noticed that the meta classifier is capable of identifying twelve different classes, eleven of which are DDoS attack classes while the other is a benign traffic.

To obtain a robust model, one that is immune from overfitting/underfitting and has a high classification accuracy as well as a low false alarm rate, a meta classifier is built based on the four base models. This meta classifier takes as an input each of the four base models then learns how each model was able to make it's own predictions. With this gained knowledge, the meta classifier can perform better and reach an accuracy higher than any single model (or base model). This also eliminates the problem of high variance in neural networks.

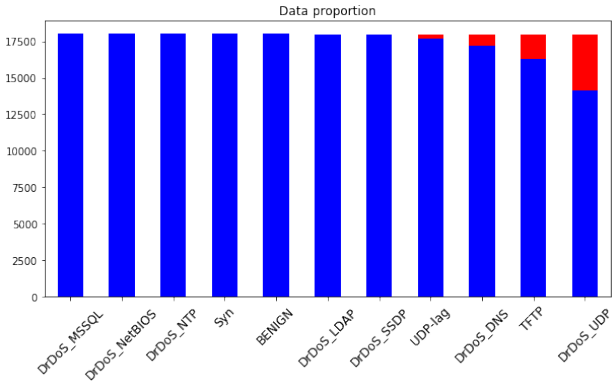
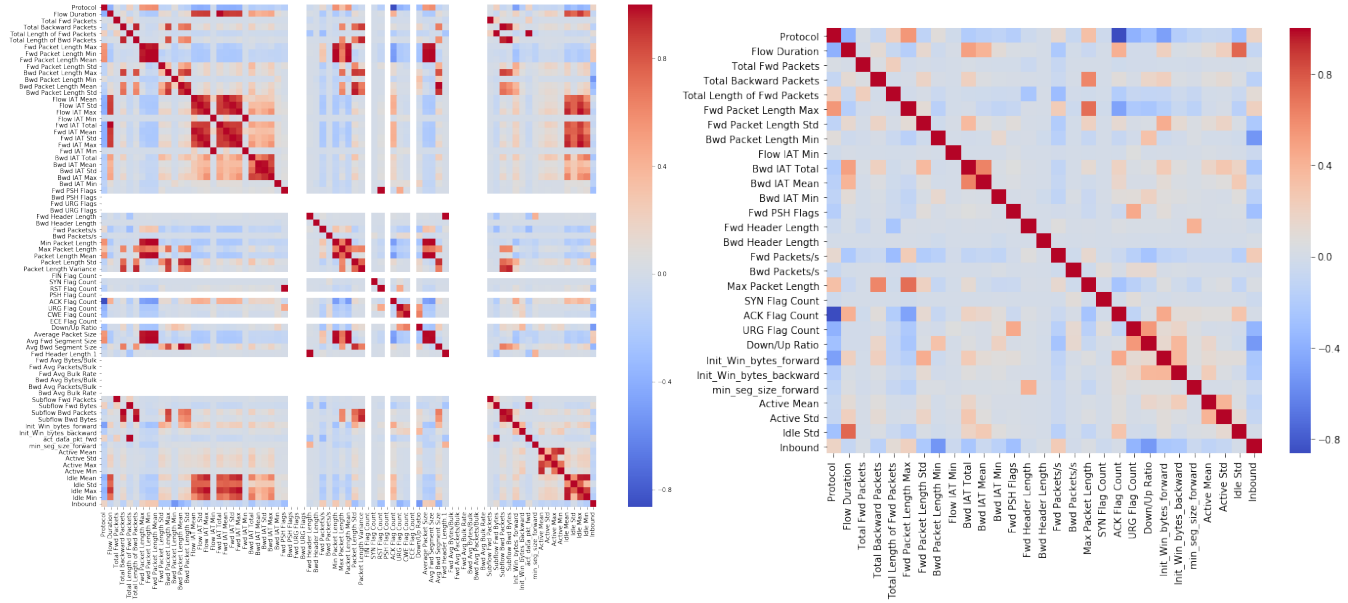


Figure 3. Data balancing using over-sampling. The red portion are the over-sampled parts of the dataset.

3. Performance Evaluation

In this section, the effects of applying phase 1 and phase 2 in the proposed scheme will be evaluated. This evaluation is done using the CICDDoS2019 [14] dataset which consists of modern day DDoS attack traffics and is to the best of the authors knowledge the latest collection and highly sophisticated DDoS attack dataset.



(a) A plot of all data showing that highly correlated and redundant parameters exists in the dataset.

(b) Resulting parameters after performing correlation test and eliminating features whose correlation score is greater than the correlation threshold.

Figure 4. Correlation heatmap of the dataset before and after eliminating highly correlated features. One has a distinct diagonal indicating that the a feature is only correlated to itself to a great extent, while the other shows features having a high correlation with other features as well as itself. **NB:** The darker the red, the higher the correlation. A feature will always have a high correlation with itself, hence the diagonal

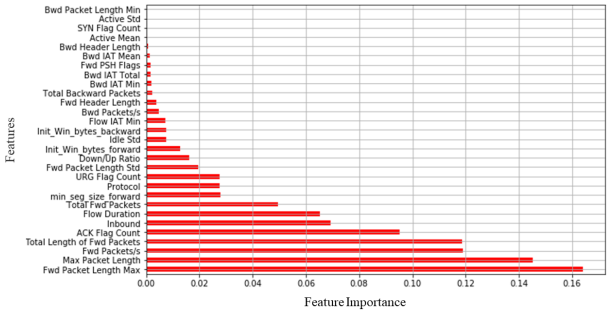


Figure 5. Feature ranking depicting features and their ranks after correlation test.

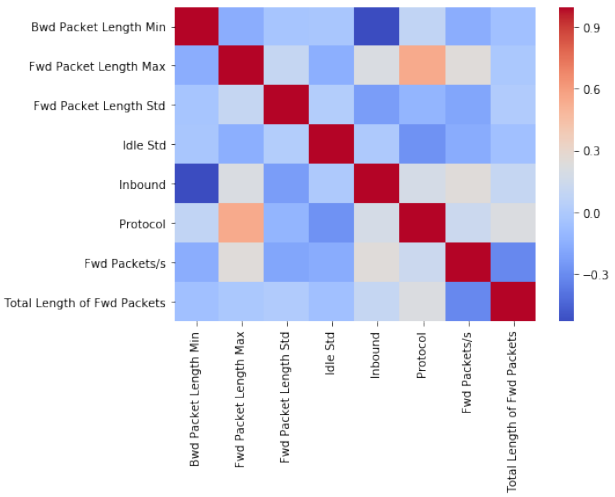


Figure 6. Correlation test of the 8 highest ranked parameters showing no correlation among parameters.

The simulations were done on google colab and keras-tuner was used in picking various hyperparameters and configuration of each base model for optimal results.

Results and discussion of each step taken will be done in an order that corresponds with the steps/order earlier presented in Fig. 1 (data balancing → correlation test → feature importance → meta classifier).

In Fig. 3. the result of implementing SMOTE for oversampling is depicted. It is seen that four classes have slightly lesser representation that others, these classes were over-sampled to prevent an uneven class representation.

The effect of correlation on the dataset is presented in form of a heatmap in Fig. 4. Fig 4a is the correlation plot of all the features in the dataset, and it clearly shows features having a very high correlation with other features. Some of this correlation is as high as the correlation the feature has with itself, and this can be interpreted as a feature appearing more than once (a redundant feature). It also shows some empty features that needs cleaning. On eliminating these redundant fea-

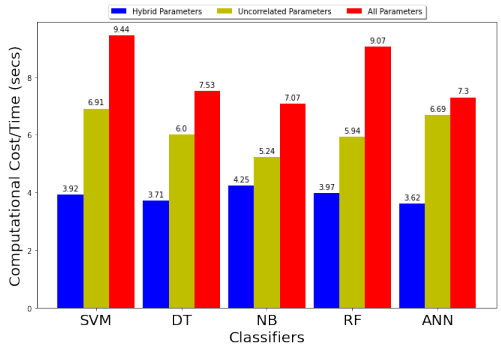


Figure 7. Testing time comparison for ML-based detectors on different feature-sets.

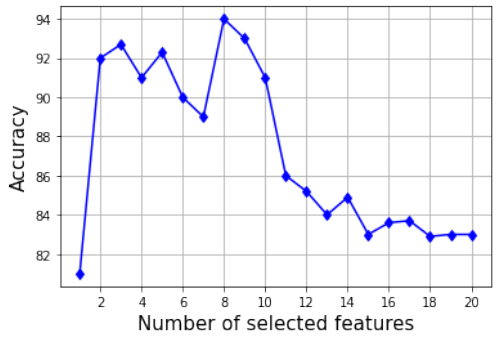


Figure 8. Test accuracy comparison for different number of highest ranked features.

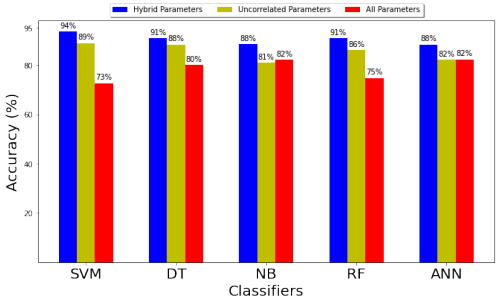


Figure 9. Test accuracy comparison for different feature-sets on four ML-based detectors.

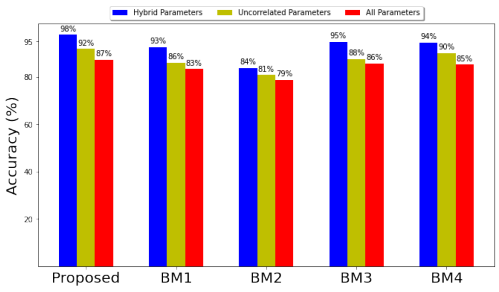


Figure 10. Average classification report of the ML-based detectors on different feature-sets.

tures as describe in Section 2, we get the heatmap in Fig. 4b. This paints a different picture from the previous heatmap as highly correlated features are no longer present. Both heatmap could be read using this simple principle :- The darker the red, the higher the correlation, and a feature will always have a high correlation with itself, this explains the dark red that is always seen on the diagonal.

As earlier mentioned, correlation on it's own does not always eliminate unwanted features. This next process ranked features based on their importance to a model. Fig. 5. shows the ranking according to importance of all features gotten from the previous stage (correlation stage). Unsurprisingly, not all features contribute to the model's output, some contribute very few, some others contribute nothing (not useful features) while there are those that contributes a lot to the model (the optimal features). It is this latter features that we are interested in, hence we have taken the eight (8) highest ranked features to be used in phase 2. To further verify how useful and uncorrelated this 8 features are, Fig. 6 gives the correlation test of the 8 optimal features, and it is clearly seen that these features have little to no correlation amongst themselves, hence they are the optimal features needed by the model

One more reason for ensuring that only optimal parameters are used in a model is that it reduces the size of the model, reduces the training/testing time and by extension reduces the computational cost of the final model. As is seen in Fig. 7., the testing time for a classifier built using all parameters is higher than when any sort of parameter selection is used. It also shows that while using uncorrelated parameters reduces the time, a hybrid parameter selection approach as used in this study still has a lesser testing time. This result as well as others to follow was collected using five different classifiers :- SVM, decision tree (DT), naive bayes (NB), random forest (RF) and artificial neural network. This is done to ensure that the classifiers had no role to play in the values of metrics that is been studied (in this case testing time).

In terms of accuracy of classification, Fig. 8. highlights that care must be taken in picking the value of $max\_no\_of\_features$. The higher the value the more likely the tendency to include features with low ranking which invariably reduces the accuracy of the model.

The effects of redundant features or lack of is shown in Fig. 9. Still using various classifiers, it is observed that the more redundant and unimportant features the model is built with, the lower the test accuracy of that model. The model might have a high training accuracy but cannot replicate that accuracy in the real world, this

is often due to overfitting or underfitting introduced by the redundant features during training. When the hybrid method is applied, test accuracy was higher than when only correlation-based method was used. The accuracy suffered drastically when no parameter selection algorithm was used.

The main objective of phase 2 was to improve accuracy and robustness. In Fig. 10 the accuracy of the four base models and that of the meta classifier (proposed scheme) is presented. The meta classifier was able to better combine the four other models predictions to make itself more accurate and robust. A look at the various feature-sets (Hybrid, Uncorrelated and All parameters) shows the same trend where the hybrid parameter set performs more than that of (only) uncorrelated parameter set and all parameters. Also, it is observed that for any of the feature-set, the meta classifier records a higher accuracy than any one of the base model. For example, the accuracy of the four base models for the hybrid feature-set are 93, 84, 95 and 94% respectively. However, the accuracy of the meta classifier for that same feature-set was 98%. It should also be noted that this study makes use of testing accuracy alone, as training accuracy can be influenced by a wide range of issues like overfitting.

4. Conclusion

In this work, a cost-efficient DDoS attack detection framework was proposed. The proposed algorithm comprises of two phases, the first phase was aimed at mitigating the curse of dimensionality by using a hybrid filter-based parameter selection approach. This approach termed '$\rho - chi - square$", involved eliminating redundant and highly correlated features and then ranking the resulting features based on their importance to the classifier to achieve a superior detection result at a minimal computational cost. In the second phase the aim was to obtain a more robust model and consisted of four base models and a meta classifier. The meta classifier is built such that it learns how to combine the predictions obtained from the four base models, hence performing better than any single model. Simulation results show that by applying the proposed filter-based approach, the DDoS detection rate is significantly improved irrespective of the machine learning algorithm used. Also, eliminating irrelevant/redundant parameters greatly improved the computational cost and brought to a minimum testing time. Furthermore, meta classifier was seen to have a higher accuracy than any of the four different base models used. By feeding the model with salient data, and implementing the stacked ensemble learning, the propose scheme was able to correctly classify the various DDoS attacks with a high accuracy.

Conflict of interest

The authors declare that there is no conflict of interest in this paper

References

[1] H. Tran-Dang, N. Krommenacker, P. Charpentier, D.-S. Kim, The Internet of Things for Logistics: Perspectives, Application Review, and Challenges, IETE Technical Review 0 (0) (2020) 1–29.

[2] H. Tran-Dang, N. Krommenacker, P. Charpentier, D. Kim, Toward the Internet of Things for Physical Internet: Perspectives and Challenges, IEEE Internet of Things Journal 7 (6) (2020) 4711–4736.

[3] S. T. Zargar, J. Joshi, D. Tipper, A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks, IEEE communications surveys & tutorials 15 (4) (2013) 2046–2069.

[4] A. S. Mamolar, Z. Pervez, Q. Wang, J. M. Alcaraz-Calero, Towards the detection of mobile ddos attacks in 5g multi-tenant networks, in: 2019 European Conference on Networks and Communications (EuCNC), 2019, pp. 273–277.

[5] Largest DDoS Attack (Accessed October 15, 2020). URL https://www.bbc.com/news/technology-53093611

[6] Understanding DDoS Attack its Effect in Cloud Environment, Procedia Computer Science 49 (2015) 202–210, proceedings of 4th International Conference on Advances in Computing, Communication and Control (ICAC3'15).

[7] D. Yin, L. Zhang, K. Yang, A DDoS Attack Detection and Mitigation With Software-Defined Internet of Things Framework, IEEE Access 6 (2018) 24694–24705.

[8] K. S. Sahoo, B. K. Tripathy, K. Naik, S. Ramasubbareddy, B. Balusamy, M. Khari, D. Burgos, An Evolutionary SVM Model for DDOS Attack Detection in Software Defined Networks, IEEE Access 8 (2020) 132502–132513.

[9] L. Zhu, X. Tang, M. Shen, X. Du, M. Guizani, Privacy-Preserving DDoS Attack Detection Using Cross-Domain Traffic in Software Defined Networks, IEEE Journal on Selected Areas in Communications 36 (3) (2018) 628–643.

[10] J. Cui, J. He, Y. Xu, H. Zhong, Tddad: Time-based detection and defense scheme against ddos attack on sdn controller, in: W. Susilo, G. Yang (Eds.), Information Security and Privacy, Springer International Publishing, Cham, 2018, pp. 649–665.

[11] V. Ravindranath, S. Ramasamy, R. Somula, K. S. Sahoo, A. H. Gandomi, Swarm Intelligence Based Feature Selection for Intrusion and Detection System in Cloud Infrastructure, in: 2020 IEEE Congress on Evolutionary Computation (CEC), 2020, pp. 1–6.

[12] S. Visalakshi, V. Radha, A literature review of feature selection techniques and applications: Review of feature selection in data mining, in: 2014 IEEE International Conference on Computational Intelligence and Computing Research, 2014, pp. 1–6.

[13] Stability of feature selection algorithm: A review, Journal of King Saud University - Computer and Information Sciences.

[14] I. Sharafaldin, A. H. Lashkari, S. Hakak, A. A. Ghorbani, Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy, in: 2019 International Carnahan Conference on Security Technology (ICCST), 2019, pp. 1–8.

Conflict of Interest

Declaration of interests

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

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: