Critical review of machine learning approaches to apply big data analytics in DDoS forensics

Kian Son Hoon[1] , Kheng Cher Yeo[2], Sami Azam[2], Bharanidharan Shanmugam[2], Friso De Boer[2]
[1]Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, Malaysia.
[2] School of Engineering and IT, Charles Darwin University, Australia.
Charles Darwin University, Australia, Darwin 0909
ldwgkshoon@gmail.com, charles.yeo@cdu.edu.au

Abstract— **Distributed Denial of Service (DDoS) attacks are becoming more frequent and easier to execute. The sharp increase in network traffic presents challenges to conduct DDoS forensics. Despite different tools being developed, few take into account of the increase in network traffic. This research aims to recommend the best learning model for DDoS forensics. To this extend, the paper reviewed different literature to understand the challenges and opportunities of employing big data in DDoS forensics. Multiple simulations were carried out to compare the performance of different models. Two data mining tools WEKA and H2O were used to implement both supervised and unsupervised learning models. The training and testing of the models made use of intrusion dataset from oN-Line System – Knowledge Discovery & Data mining (NSL-KDD). The models are then evaluated according to their efficiency and accuracy. Overall, result shows that supervised learning algorithms perform better than unsupervised learning algorithms. It was found that Naïve Bayes, Gradient Boosting Machine and Distributed Random Forest are the most suitable model for DDoS detection because of its accuracy and time taken to train. Both Gradient Boosting Machine and Distributed Random Forest were further investigated to determine the parameters that can yield better accuracy. Future research can be extended by installing different DDoS detection models in an actual environment and compare their performances in actual attacks.**

Keywords--DDoS, Cyber forensics, Big Data, Security, DDoS detection

I. INTRODUCTION

In the current era, Internet based applications are the driving force. But, the architectural vulnerabilities have made it easier for malicious attackers to conquer the Internet. It is very obvious that securing the network is an ongoing process. The recent attacks [19] on financial institutions and other government agencies proves that DDoS attack is an ongoing threat and needs to be handled with a cautious approach. It is now well established from a variety of studies that Distributed Denial of Service (DDoS) analysis is increasingly challenging due to the increase in the network flow [1- 4]. Anstee [5] explained that the DDoS are evolving and there is a prevalence of advanced DDoS, which can circumvent current defense mechanism. They further discussed that a lack of preparation against DDoS attacks will be costly to any organisation if they have become a target. Researchers Zowoad and Hasan [4] explain the large-scale potential evidence also makes it harder to detect anomalies in the data. Khattak et al. [2] noted the difficulty in doing DDoS forensics due to the size of the log file and the traditional methods used to mine the data.

In contrast, the challenges and opportunities presented by big data have been discussed comprehensively by number of authors [2, 3, 4, 6, 7, 8]. In order to address the issue, Lee and Lee [3], revealed that some tools have been developed to aid in Internet traffic analysis and DDoS forensics. The tools include but are not limited to Tcpdump, libpcap, Wireshark, CoralReef, and Snort [3]. However, the researchers also mentioned that these tools are inadequate and do not scale well with the massive data. Different methods to do DDoS forensics have been suggested, not limited to cluster analysis [9], mapReduce [2] and neural network [6]. Different researchers have implemented different big data algorithms to do DDoS forensics. Previous researches has compared different supervised learning models in detecting DDoS and differentiating the attacks in DARPA 98 dataset using Waikato Environment for Knowledge Analysis (WEKA). So far, however, there has been little discussion about performances of different unsupervised learning models to detect DDoS in Defense Advanced Research Project Agency (DARPA) 98 dataset. Moreover, research on the subject has been mostly restricted to the usage of DARPA 98 or oN-Line System – Knowledge Discovery & Data mining (NSL-KDD) 99 dataset.

The objective of this research is to determine whether supervised or unsupervised machine learning models may be suitable for DDoS detection. The research will analyse the performance of different machine learning algorithms. Finally, the performances of different parameters for 2 different algorithms (identified at the earlier stage to be the best) are also compared.

II. LITERATURE REVIEW

A. Current DDoS Forensics methods

Prasad et al. [8] had classified DDoS detection methods as follows: (1) statistical, (2) knowledge based, (3) soft computing, and (4) data mining & machine learning. Alternatively, Prasad [4] categorised the different methods

978-1-5386-2238-4/18/$31.00 ©2018 IEEE

according to their working mode, i.e. centralised and distributed. Different ways to categorise detection methods were mentioned by Douligeris and Mitrokotsa [10] and Peng[11]. They categorise them into DoS-specific/ misuse detection and anomaly detection. It is important to note that a method can be categorised in more than one way, e.g. an algorithm can be categorised as a centralised soft computing algorithm that is DoS-specific. A summary of the characteristics of the classification is listed in table 1.

TABLE I. SUMMARY OF DIFFERENT CLASSES OF DDOS DETECTION METHODS [8,9]

Class	Description
Statistical	Exploiting the statistical properties of normal and attack patterns. Normal traffic is modelled statistically and statistical inference test is applied to new instances of traffic to determine if they are normal.
Knowledge based	Known attacks are identified and their signatures are used to identify actual occurrences of attacks.
Soft computing	Using learning paradigms that are optimised and tolerant of imprecision and uncertainty.
Data mining and machine learning	Using automated extraction of hidden predictive information from network data to build a model which can be used to classify new network traffic.
DoS-specific / misuse detection	Identifying the intrusion pattern of known DoS attacks by analysing their characteristics. The pattern is then used to generate signatures and can be used to detect DoS.
Anomaly detection	Model the behaviour of normal traffic and use the model to detect Dos attack.
Centralised mode	Detection and prediction are done on a machine.
Distributed mode	Detection and prediction are done on multiple machines.

Karim et al. [12] mentioned that there is a growing trend of using data mining and machine learning detection methods, e.g. support vector machine, neural network, and cluster analysis. Moreover, there is also an increased interest in doing DDoS detection using distributed mode[3,13]. In summary, the current trend of doing DDoS detection takes into account of the increase of traffic and the insight that can be generated by the availability of data.

B. Applying big data analytics in network forensics

There are two perspectives in doing big data forensics [4]. First is finding a small piece of evidence that exists in a big dataset and using big data to discover new or unknown facts [4]. An example of the first perspective is finding out the identity of the hacker. On the other hand, the latter perspective can be seen in using big data to train a more effective spam mail filter.

In terms of DDoS forensics, emphasis is placed on the second perspective. The second perspective is also the main focus of this literature review. Numerous researches demonstrated that big data can be used to improve the intrusion detection system [14-17], Noureldien & Yousif [14]; Sabhanani & Serpen [15]; Nguyen & Choi [16]; Jalil et al. [17]). On the other hand, an example of the first perspective is amalgamating data from different machines are used to pinpoint the exact perpetrator. Despite more researches dedicated to the second perspective, it is interesting to see that both approaches are useful in DDoS forensics.

A non-exhaustive list of the big data algorithms that have been applied in the context of DDoS forensics by different researches are summarised in table II. Most of the researchers also utilised the WEKA, a suite of machine learning algorithms developed at University of Wakaito [14-17]. More details on the various algorithms, including the mathematical expressions can be found in the respective papers shown in the table below.

TABLE II. SUMMARY OF ALGORITHMS APPLIED IN CONTEXT OF DDOS FORENSICS

Algorithms	Researchers
Radial Basis Function neural network	Karimazad & Faraahi (2011) [6]
K-Nearest Neighbour	Nguyen & Choi (2010) [16]
PART, BayesNet, IBK, Logistic, J48, Random Committee, Input Mapped Classifier	Noureldien & Yousif (2016) [14]
Multilayer perceptron (neural network), Gaussian classifier, K-means clustering, Nearest cluster algorithm, Incremental Radial Basis Function, Leader algorithm, Hypersphere algorithm, Fuzzy Adaptive Resonance Theory Mapping, C4.5 decision tree	Sabhanani & Serpen (2003) [15]
Neural Network, Support Vector Machine, Decision Tree (J48)	Jalil et al. (2010) [17]

C. Performance measures of learning algorithms

Caruana 2003 [18] stated that recall; precision and accuracy could be used to measure the performance of different machine learning algorithms. The terms defined by the researchers are summarised in table below (Table III).

TABLE III. PERFROMANCE MEASURE AND DEFINITION

Term	Definition
Accuracy	Proportion of correct result achieved; (TP+TN)/(TP+TN+FP+FN) TP: True Positive, TN: True Negative FP: False Positive, FN: False Negative
Precision	Proportion of returned positive that is actually positive; TP/(TP+FP)
Recall	Proportion of actual positives found; TP/(TP+FN)
F1	Balancing tradeoffs between Precision and Recall; 2(Precision*Recall) / (Precision+Recall)

III. METHODOLOGY

Simulation was adopted to compare the accuracy and efficiency of different models in detecting DDoS. Many researchers have utilised simulation to measure the accuracy of different learning models in detecting DDoS attacks in the past. They do so by calculating the percentage of true positive, true negative, false positive and false negative [3,6,9,14-15]. In terms of efficiency, computing time (both training and testing) can be a good indicator of the performance of a model. In such a case, the machine used to do the training and

testing should be consistent in order to ensure that each simulation is executed under similar circumstances.

The benefit of this approach is that it simplifies the complexity of DDoS attacks. Simulation helps to simplify the data gathering process where instead of capturing real net flow data, dataset available online can be used. Simulated past attack net flow can also be used to test the trained model instead of setting up a network and launching DDoS attack. The limitation of this approach is that it may over-simplify the actual situation. For example, the dataset used was already pre-processed before training and testing. However, this is not the case in an actual situation. The overhead generated by the higher features (features that are derived from normal features) from the data may incur overhead. Moreover, the captured dataset may not necessarily reflect the constantly changing actual threat environment.

The dataset that has been used in this study is oN-Line System – Knowledge Discovery & Data mining (NSL-KDD) dataset which is available online. The dataset contains a large number of net flows. The data are classified as either normal or abnormal. The data is ready to be fed into the data learning algorithms in its original form because it has been pre-processed and are labelled.

Waikato Environment for Knowledge Analysis (WEKA) and H2O are used in this study. Both software is popular and has been used extensively in data science. WEKA is used because of the visualisation ability. Furthermore, WEKA is also used to train and validate the unsupervised machine learning algorithms, which is not available on H2O. On the other hand, H2O is used for training and testing the supervised learning algorithms. H2O is used because of its optimised algorithms, which can perform faster than WEKA.

IV. RESULTS

The results are arranged in 3 sections, various software's are compared in section A; performances of supervised and unsupervised learning algorithms are compared in section B; and the parameters that return the most suitable model and threshold returning most accurate prediction is presented in section C.

A. Software comparison

Before deciding the software to be used for this research, different software has been explored. This software includes H2O, Orange, Rattle and WEKA. The characteristics and support of the software are summarised in Figure 1. All of the mentioned software contains libraries that can be extended and used in programming. They are chosen because the software also consists of graphics user interface (GUI) which eases the task.

Software	Usability	Visualisation	Open source	Language	Documentation	Community of users	Scalability
WEKA	Easy	Yes	Yes	Java	Yes	Yes	No
Orange	Not easy	Yes	Yes	Python	Yes	Yes	No
Rattle	Easy	Yes	Yes	R	Yes	Not as much	Yes
H2O	Easy	No	Yes	Java, Python, R	Yes	Yes	Yes

Figure 1. Comparison of Different Data Mining Tools

Orange requires Python to run on and provides "Orange.Canvas" as the GUI. Similarly, Rattle requires R programming environment and provides its on GUI with the 'Rattle()' commands. Rattle is a relatively new project and it is slowly gaining popularity. WEKA provides its own GUI through in windows and is one of the most commonly used software both by researchers in evaluating the performance of different learning models as well as other data science students. The most significant feature about WEKA is that it contains a large number of learning algorithms described by different researchers. Moreover, the visualisation capability of WEKA also makes it popular. In contrast, H2O offers only a limited number of highly optimised learning algorithms. Another advantage of H2O is that it gives a good summary of the performance of the models at different threshold. H2O provides its GUI through H2O flows in browser.

After testing the different software, H2O and WEKA are chosen for analysis. The primary advantage of choosing H2O over WEKA is that all the algorithms that it has to offer can be tweaked easily. As mentioned, the algorithms in H2O are also optimised and usually perform faster than that in H2O given similar parameters. WEKA is also chosen primarily because of its visualisation capability and because of its ability to validate and evaluate unsupervised learning models.

B. Comparision of supervised and unsupervised learning algorithms

The performances of different algorithms (both supervised and unsupervised) have been compared. All the parameters' value has been set to default supplied by H2O. Moreover, the results are also obtained in the same machine. Fig. 1 shows the graph of accuracy versus time taken to train the different models with outliers removed.

It can be seen that supervised learning models (deep Learning, distributed random forest, gradient boosting machine and naïve bayes) generally perform better than unsupervised learning models (canopy, farthest first, filtered cluster and make density based cluster). Among the unsupervised learning models, Make Density Based Cluster is the most accurate, followed by Farthest First. Gradient Boosting Machine (GBM), Naïve Bayes and Distributed Random Forest (DRF) are the most accurate amongst supervised learning models.

Gradient Boosting Machine and Distributed Random Forest are selected for further fine-tuning to determine if the algorithms can perform better with other parameters. Naïve Bayes is not selected because the algorithm does not contain any modifiable parameter. The Gradient Boosting Algorithms of H2O is as follows [20]:

1. Initialise $f_{k0} = 0, k = 1,2, \dots, K$
2. For m = 1 to M:
   a. Set $p_k(x) = \frac{e^{f_k(x)}}{\sum_{l=1}^{K} e^{f_l(x)}}, k = 1,2, \dots, K$
   b. For k = 1 to K:
      i. Compute $r_{ikm} = y_{ik} - p_k(x_i), i = 1,2, \dots, N$
      ii. Fir a regression tree to the targets $r_{ikm}, i = 1,2, \dots, N$, giving terminal regions $R_{jim}, j = 1,2, \dots, J_m$
      iii. Compute $\gamma_{jkm} = \frac{K-1}{K} \frac{\sum_{a_i \in R_{jkm}} (r_{ikm})}{\sum_{a_i \in R_{jkm}} |r_{ikm}|(1-r_{ikm})}, j = 1,2, \dots, J_m$
      iv. Update $f_{km}(x) = f_{k,m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jkm} I(x \in R_{jkm})$
3. Output $\hat{f}_k(x) = f_{kM}(x), k = 1,2, \dots, K$

The algorithms for Distributed Random Forest can be found at [21]. Performance of the supervised learning algorithms depend very much on the training data and are less flexible than unsupervised learning algorithms. This is because supervised learning algorithms make predictions based on a set of examples. If the number of examples given is not enough or when the examples fail to cover different instances of the problems, the supervised learning algorithms may perform badly.

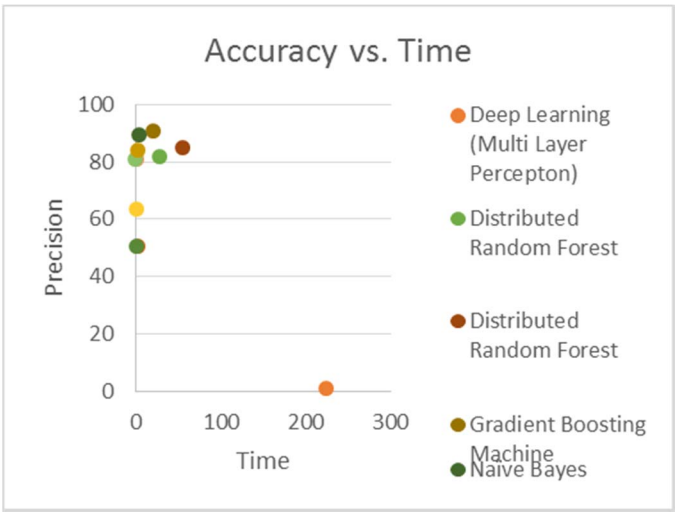

Figure 2. Accuracy vs. time to train.

C. *Investigating parameters for model and threshold for prediction*

The tables (Table IV and V) below shows the best parameters and prediction threshold identified for both Gradient Boosting Machine and Distributed Random Forest.

TABLE IV. BEST PARAMETERS & PREDICTION THRESHOLD FOR GRADIENT BOOSINT MACHINE

Number of trees	51
Depth	5
Learning Rate	0.10
Threshold	0.993

TABLE V. BEST PARAMETERS & PREDICTION THRESHOLD FOR DISTRIBUTED RANDOM FOREST

Number of trees	500
Depth	20
Sampling Rate	0.10
Threshold	0.9847

The steps taken to find the best parameter are as followed: (1) beginning with the defaults parameters provided by H2O, find the right number of trees; (2) set the number of trees to that which provides the highest accuracy and compute the depth; (3) set the depth to that which provides the highest accuracy and calibrate the learning rate (Gradient Boosting Machine) or sampling rate (Distributed Random Forest).

In the step (2), the performances of different models are compared using their maximum accuracy, these models all have high maximum precision and high maximum recall, ensuring that models have certain predictive power. Gradient Boosting Machine has 0.9075, 0.9972, and 1.0 as maximum accuracy, precision and recall respectively. On the other hands, Distributed Random Forest has 0.9326, 0.9987, and 1.0 as the maximum accuracy, precision and recall respectively. The time taken to train the models are 12.221s (for Gradient Boosting Machine) and 129.128s (for Distributed Random Forest) with the stated parameters using the same machine. The time taken to build the models using a normal laptop is short enough and should enable a cyber security professional to utilize H2O software to conduct digital forensics.

The prediction task will make use of the model and threshold value to classify a connection either as normal or abnormal. Formally, if P(A) > threshold, the connection is classified as abnormal, where P(A) is the probability of the connection being an abnormal connection. Suitable threshold for prediction within the model is identified using F1 score. F1 score helps to decide the tradeoff between precision and recall. This ensures that the prediction returns as many anomalies and as accurate as possible. The highest F1 score achieved by Gradient Boosting Machine is 0.8808 while Distributed Random Forest achieved score of 0.9847.

V. CONCLUSION

This research sheds new light on the performance of different learning models in detecting DDoS attack as well as

how the parameters will affect the performance of the models. Algorithms were implemented using H20 and a comparison of the accuracy of the algorithms in detecting DDoS attacks was carried out.

Compared to literature, this research considers the use of big data analytics from the point of views of a security analyst with minimal exposure to data analytics. For this reason, different available software has been compared. Moreover, this research also adds on comparing more learning algorithms and finds their usefulness. Another key importance of this research is the usage of H2O instead of WEKA. The main argument for this is that H2O is more suitable for the industry as opposed to WEKA due to the highly optimised algorithms. Algorithms in H2O are also more flexible than that in WEKA and can be customised easily.

ACKNOLWDGEMENTS

The authors would like to thanks School of Engineering and IT, Charles Darwin University for funding this research publication.

REFERENCES

[1] O. M. Adedayo, "Big data and digital forensics," 2016 IEEE International Conference on Cybercrime and Computer Forensic (ICCCF), Vancouver, BC, 2016, pp. 1-7.

[2] R. Khattak, S. Bano, S. Hussain and Z. Anwar, "DOFUR: DDoS Forensics Using MapReduce," 2011 Frontiers of Information Technology, Islamabad, 2011, pp. 117-120.

[3] Y. Lee and Y. Lee, "Toward scalable internet traffic measurement and analysis with Hadoop", ACM SIGCOMM Computer Communication Review, vol. 43, no. 1, p. 5, 2012.

[4] S. Zawoad and R. Hasan, "Digital Forensics in the Age of Big Data: Challenges, Approaches, and Opportunities," 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems, New York, NY, 2015, pp. 1320-1325.

[5] D. Anstee, "Preparing for tomorrow's threat landscape", *Network Security*, vol. 2015, no. 8, pp. 18-20, 2015.

[6] R. Karimazad and A. Faraahi, "An Anomaly-Based Method for DDoS Attacks using RBF Neural Networks" in 2011 International Conference on Network and Electronics Engineering, Singapore, 2011, pp. 44-48.

[7] G. No and I. Ra, "An efficient and reliable DDoS attack detection using a fast entropy computation method," 2009 9th International Symposium on Communications and Information Technology, Icheon, 2009, pp. 1223-1228.

[8] K. Prasad, A. R. M. Reddy and K. V. Rao, "DoS and DDoS attacks: defense, detection and traceback mechanisms - a survey," Global J. of Computer Science and Technology: E Network, Web & Security, vol. 14, no. 7, pp. 15-32.

[9] K. Lee, J. Kim, K. Kwon, Y. Han and S. Kim, "DDoS attack detection method using cluster analysis", Expert Systems with Applications, vol. 34, no. 3, pp. 1659-1665, 2008.

[10] C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: classification and state-of-the-art", Computer Networks, vol. 44, no. 5, pp. 643-666, 2004.

[11] T. Peng, C. Leckie and K. Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems", ACM Computing Surveys, vol. 39, no. 1, p. 3-es, 2007.

[12] A. Karim, R. Salleh, M. Shiraz, S. Shah, I. Awan and N. Anuar, "Botnet detection techniques: review, future trends, and issues", J. of Zhejiang University SCIENCE C, vol. 15, no. 11, pp. 943-983, 2014.

[13] B. B. Gupta, R. C. Joshi, and M. Misra, "ANN Based Scheme to Predict Number of Zombies in a DDoS Attack," Int. J. of Network Security, vol. 14, no. 2, 2012, pp. 61-70.

[14] A. N. Noureldien and I. M. Yousif, "Accuracy of machine learning algorithms in detecting DoS attack types", Journal of Science and Technology, vol. 6, no. 4, pp. 89-92, 2016.

[15] M. Sabhnani and G. Serpen, "Application of machine learning algorithms to KDD intrusion detection dataset within misuse detection context" International Conference on Machine Learning; Technologies and Applications (2003), Las Vegas, NV, 2003, pp. 209-215.

[16] H. Nguyen and D. Choi, "Application of data mining to network intrusion detection: classifier selection model," 11th Asia-Pacific Network Operations and Management Symposium, Beijing, 2008, pp. 399-408.

[17] K. A. Jalil, M. H. Kamarudin and M. N. Masrek, "Comparison of Machine Learning algorithms performance in detecting network intrusion," 2010 International Conference on Networking and Information Technology, Manila, 2010, pp. 221-226.

[18] R. Caruana and Computer Science Department Cornell University (2003,). *Performance measures for Machine Learning* [Online]. Available: https://www.cs.cornell.edu/courses/cs578/2003fa/performance_measures.pdf

[19] CDNetworks Q2 2017 DDoS Attack Trends Report https://www.cdnetworks.com/sg/resources/CDNetworks_DDoS%20Attack%20Trends_Q2%202017_ENG_final_20170821-2-.pdf. Last accessed October 2017.

[20] Hastie, Trevor, Robert Tibshirani, and J Jerome H Friedman. The Elements of Statistical Learning. Vol.1. N.p., page 339: Springer New York, 2001.

[21] Breiman, L. Machine Learning (2001) 45: 5. https://doi.org/10.1023/A:1010933404324