# A Deep Learning Approach for DDoS Attack Detection Using Supervised Learning

***Hailye Tekleselassie***
***Department of Information Systems, School of Informatics, Wolaita Sodo University, Wolaita Sodo, Ethiopia***

**Correspondence should be addressed to Hailye Tekleselassie; hailyie.tekleselase@wsu.edu.et**

**Abstract**
This research presents a novel combined learning method for developing a novel DDoS model that is expandable and flexible property of deep learning. This method can advance the current practice and problems in DDoS detection. A combined method of deep learning with knowledge-graph classification is proposed for DDoS detection. Whereas deep learning algorithm is used to develop a classifier model, knowledge-graph system makes the model expandable and flexible. It is analytically verified with CICIDS2017 dataset of 53,127 entire occurrences, by using ten-fold cross validation. Experimental outcome indicates that 99.97% performance is registered after connection. Fascinatingly, significant knowledge ironic learning for DDoS detection varies as a basic behavior of DDoS detection and prevention methods. So, security professionals are suggested to mix DDoS detection in their internet and network.

***Key words:***
*Distributed denial of Service; wireless networks; deep Learning Algorithms; Transmission Control Protocol; CNN; network security*

## 1. Introduction

The increase of IoT devices and computation devices have completed living relaxed and suitable for us due to the debauched and correct computation of our information. But, augmented incorporation and placement of linked devices also disclosures vital capitals to DDoS threats [1].
Technological growths in current years have complete it likely to connect a variety of devices to computer networks, which brings various benefits to users. But, with the increase of the technologies elaborate, the number of cyberattacks is also increasing, using more sophisticated means to incorrectly access sensitive information and to extort money or the already mentioned interruption of services. One such technology is the Internet of Things (IoT) [2].
The idea of the Internet of Things includes various devices, sensors, objects, and intelligent nodes that are able to function autonomously and communicate with each other without human intervention. Such IoT devices are able to deliver a number of valuable facilities and, cheers to sensors and actuators, provide various data in real-time. In many cases, however, devices in the field of the Internet of Things, in particular, contain various software bugs brought in from the factory that make them vulnerable. Such vulnerabilities often allow attackers to perform various cyberattacks and compromise the security of the environment in which IoT devices are located [3].
Several defense mechanisms have been proposed in the past against DDoS attacks in IoT networks. They can be divided into two basic groups: traditional DDoS defenses and IoT-specific DDoS defenses. They fluctuate in terms of place and difficulty. While traditional DDoS defenses are applied to the target server and are fundamentally homogeneous, IoT-specific DDoS defenses are applied to IoT devices and are more complex, reflecting the heterogeneity of IoT devices. In both cases, detection techniques are used to detect abnormal activities in the network or host [4].
The rest of the paper is organized as follows: Section 2 presents related works; Section 3 elaborates the data set and describes the methodology followed in the research; Section 4 details the experimentation procedures, the result gotten and the observations from the results while Section

5 presents the conclusion of the work as well as highlighting the future work

## 2. Related works

According to [5] detection systems of network intrusion have traditionally been rule-based. Nevertheless, machine learning and statistical approaches have also made major contributions [5]. Machine learning have also proven to be effective in two main aspects of network security which are: feature engineering (i.e., the ability to extract the most important features from network data to assist model learning) [6] and classification. In security environment, classification tasks usually involve training both suspicious and benign data in order to create models that can detect known attacks [7].

The authors in [8] pointed out that steps such as collection of network information, feature extraction and analysis, and classification detection provide a means for building efficient software-based tools that can detect anomalies such as software-defined networking (SDN). Another study [9] provides a thorough classification of DDoS attacks in terms of detection technology. The study also emphasizes how the characteristics of the network security of an SDN defines the possible approaches to setting up a defense against DDoS attacks. Similarly, [10] have explored this area too. In other approaches to DDoS defense, [4] propose a scheduling based SDN controller architecture to effectively limit attacks and protect networks in DoS attacks.

The growth of cloud computing and IOT has inevitably led to the migration of denial-of-service attacks on cloud computing devices as well. Thus, cloud computing devices must implement efficient DDOS detection systems in order to avoid loss of control and breach of security [11]. Studies such as [12] aim to tackle this problem by determining the source of a DDOS attack using Trace (powerful trace) source control methods. Trace controlled such attack sources from two aspects, packet filtering and malware tracing, to prevent the cloud from becoming a tool for DDoS attacks. Other studies such as [13] approach the problem of filtering by using a set of security services called filter trees. In the study, XML and HTTP based DDOS attacks are filtered out using five filters for detection and resolution. Detection based on classification has also been proposed and a classifier system for detection against DDOS TCP flooding attacks was created [14].

These classifiers work by taking in an incoming packet as input and then classifying the packet as either suspicious or otherwise. The nature of an IP network is often susceptible to changes such as the flow rate on the network and in order to deal with such changes, self-learning systems have been proposed that learn to detect and adapt to such changes in the network [15].

Many of the existing models for DDoS detection have primarily focused on SYN-flood attacks and haven't been trained to detect botnet attributes. More studies are thus needed where models are trained to detect botnet as botnet becomes the main technology for DDoS organization and execution [16]. Botnet DDoS attacks infect multitude of remote systems turning them to zombie nodes that are then used for distributed attacks. In detecting botnet DDoS attacks, authors in [17] used a deep learning algorithm to detect TCP, UDP and ICMP DDoS attacks. They also distinguished real traffic from DDoS attacks, and conducted in-depth training on the algorithm by using real cases generated by existing popular DDoS tools and DDoS attack modes. Also, [18] proposed a DDoS attack model and demonstrated that by modelling different allocation strategies. The proposed DDoS attack model is applied to game planning strategies and can simulate different botnet attack characteristics.

According to [19] "DDoS detection approaches can operate in one of the following three modes: supervised, semi-supervised and unsupervised mode. For the detection approach in supervised mode, it requires a trained dataset (or a classifier) to detect the anomalies, where the trained dataset includes input variables and output classes. The trained dataset is used to get the hidden functions and predict the class of input variables (incoming traffic instances). This mode is similar to a predictive model. For example, Classification techniques comes under the category of supervised data mining" [20]. "For the Approaches that work in the semi-supervised mode, they have incomplete training data i.e., training data is only meant for normal class and some targets are missing for anomaly class" [21]. Unlike supervised and semi-supervised learning, unsupervised machine learning algorithms do not have any input-output pairs but the algorithm is trained such that it can accurately determine the unknown data point. The following subsections further discusses the unsupervised learning algorithms we used in this work.

current effort that goes to detect IoT based attacks proposed MQTT transaction-based features Mustafa et al. (2019). But the authors used features based on the TCP protocol analysis, which do not provide sufficient information on the MQTT protocol parameters. In contrast, our proposed UDP features are based on unsupervised machine learning which can successfully detect and distinguish such attacks including the unknown attacks.

### 2.1 Restricted Boltzmann Machines (RBM)

The authors in [24] stated that Boltzmann machine (BM) is a bidirectionally connected network of stochastic processing units. BMs are commonly used to learn important features of an unknown probability distribution based on samples from the distribution. However, the training process of the BM is usually computationally intensive and tedious. The restricted Boltzmann machine attempts to solve the training problem of BMs by imposing key restrictions on the architecture of the BM.

The BM is a fully connected network of bidirectional nodes where each node is connected to every other node. The RBM on the other hand is presented as a relatively smaller network of bidirectional nodes with the restriction that nodes on the same layer are not connected to each other horizontally [24].

The restricted Boltzmann machine is a generative model that is used to sample and generate instances from a learned probability distribution. Given the training data, the goal of the RBM is to learn the probability distribution that best fits the training data. The RBM consists of m visible units V= (V_1, V_2…., V_ m) and n hidden units H= (H_1, H_2…., H_ n) arranged in two layers.

The visible units lie on the first layer and represent the features in the training data (see Figure 2). Usually, one visible unit will represent one feature in an example in the training data. The hidden unit's model and represent the relationship between the features in the training data. The random variables (V, H) take on values (v, h) ∈ [0,1] ^m for continuous variables and the underlying probability distribution in the training data is given by the Gibbs distribution p (v, h) = 〖1/z〗 ^ (-E (v, h)) with the energy function in equation 1;

$$E(\mathbf{v},\mathbf{h}) = -\sum_{m}^{n}\sum_{m}^{n} w_{ij} h_i v_j - \sum_{j=1}^{m} b_j v_j - \sum_{i=1}^{n} c_i h_i \qquad (2)$$

In equation 2, w_ij are real valued weights associated with v_ j and h_ i, and b_ j and c_ i are real valued bias terms associated with units i and j respectively. The contrastive divergence learning algorithm is one of the successful training algorithms used to approximate the log-likelihood energy gradient and perform gradient ascent to maximize the likelihood [24].

After a successful training, the RBM should be able to represent the underlying probability distribution of the training data and when presented with unseen examples, the RBM should be able to generate similar representations to the example provided.

### 2.2 K-Means

The K-means algorithm takes the full dataset consisting of multiclass data points, then clusters the datapoints into separate clusters to the best of its ability; this classification occurs when you feed in the input and the model assigns the input into one of the computed clusters. Given a set of observations (x_1, x_2, x_3, ..., x_ n), where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into k (≤ n) sets S = {s_1, s_2, ..., s_ k} so as to minimize the within-cluster sum of squares (WCSS) (i.e., variance).

### 2.3Expectation-Maximization (EM)

The authors in [25] stated that EM algorithm is used for solving mixture models that assume the existence of some unobserved data. Mathematically, the EM algorithm can be described as follows; given the statistical model that generates a set of observed data X, latent data Z, unknown parameters $\theta$ and the likelihood function L ( $\theta$ ; X) =p (X, Z ⫞| θ), the maximum likelihood of the unknown data $\theta$ is determined by maximizing the marginal log-likelihood of the observed data X using equation 3:

In the expectation step, the likelihood of the unknown parameters is computed as the log-likelihood of the known parameter estimates, while in equation 4 the maximization step is used to select the new value that maximizes the log-likelihood given the estimates from equation 5.

In our research, we differ from current work (e.g. [26]) in two ways. First, we work with unsupervised machine learning methods using both normal and suspicious network data to train. Second, we made use of dimension reduction methods such as K-means clustering with PCA, Expectation maximization, Restricted Boltzmann Machine and Autoencoder (where K-means and EM are both trained using normal and suspicious data; RBM and AE was trained on only suspicious data), all these methods were not only for feature engineering [22] but for classification as well.

### 3. CICIDS 2017 Dataset

Currently, CICIDS 2017 is a typical intrusion dataset used in several intrusions detection design and implementation research works for justifying DDoS [30]. CICIDS 2017 dataset is an enhancement of CICIDS 2017 with a basic modification made to solve the difficulty and problems found in previous CICIDS 2017; but, still there is a problem in the new version of CICIDS 2017 however with great advantages over. CSE-CIC-IDS-2018 dataset as stated by Talwar and Goyal [5], this version of the dataset has been more applicable for real networks as well. As claimed by Aggarwal and Sharma [30], the new version of CSE-CIC-IDS-2018 modified and developed from the fundamental problem existed in the old CSE-CIC-IDS-2018 benchmark intrusion dataset. 'e problem of redundancy and missing values existed in CSE-CIC-IDS-2018 is alleviated in new CSE-CIC-IDS-2018 benchmark intrusion data set. In this empirical study, CSE-CIC-IDS-2018 intrusion dataset which is similar with CSE-CIC-IDS-2018 dataset with 16,000,000 attributes is used.

## 4. METHODOLOGY

In this research, an expandable and flexible deep learning network intrusion detection system is presented. The system is planned by mixing machine learning model with knowledge graph. method and steps followed in this research are defined as follows.

4.1. An expandable and flexible Deep Learning method. Literature on building a predictive model for distributed denial of service attack (DDoS) is rich, but the advanced DDoS does not cover the expandability and flexible behavior of the intrusion detection model. Scalability is be- coming increasingly required for today's network intrusion detection [17]. This is because of the rapid growth of the large volumes of modern network traffic that requests urgent monitoring with a repeatedly altering attack activity. In the interim, the novel method regulates and familiarizes itself with the newly updated network connections. Therefore, the deep learning robotically learns the novel difficulty while there is alteration in network connection behaviors.
The execution for the proposed method is directed with the help of Python programming language and WEKA 3.9 machine learning tool, and WEKA library functions are used for feature selection and classifier building methods.

The future method for expandable and flexible network intrusion detection is offered in Figure 1. It contains of two major modules. So, in this section, we tried to discuss the details of the proposed approaches
The BASHLITE dataset consists of 110,000 SYN-flood instances and 100,000 UDP-lag attacks. Both Mirai and BASHLITE are open-source malware that can be used for academic research purposes.

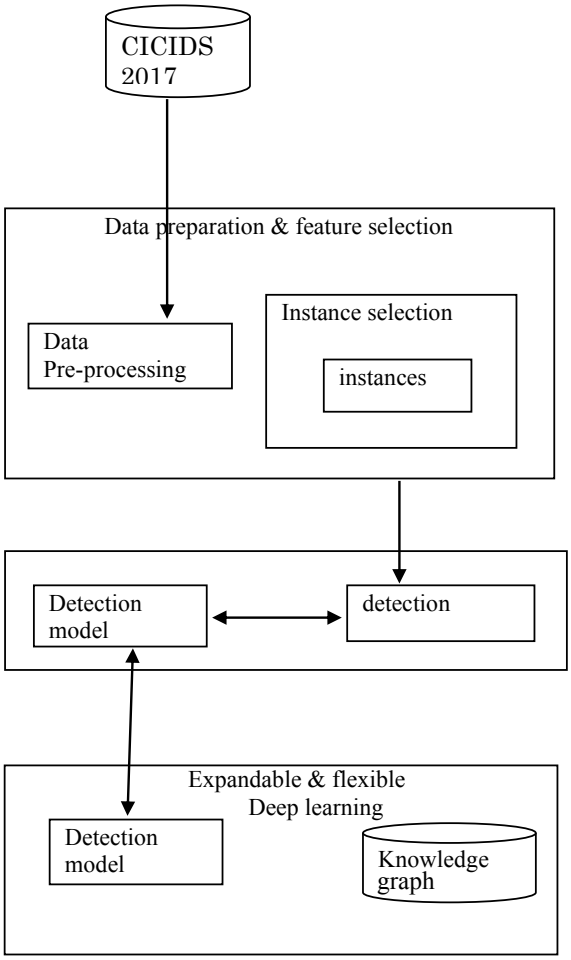



Figure 1: expandable & flexible deep leaning for DDoS

Figure 1 indicates the architecture of novel network
intrusion detection model representing its main modules and subsystems. As described in Figure 1, the novel method
the establish of two main subsystems: the supervised Vector Space Model (VSM)), connector, and the knowledge-graph system (KGS). In fact, the learning subsystem is a cooperative outcome of database, pattern extraction, and update detection modules. The learning subsystem is mainly responsible for learning from the dataset incrementally and adaptively using ma- chine learning algorithm. On the other hand, the knowl edge-graph system signifies the deep learning outcome to detect the type of incoming network traffic, and it robotically updates the novel network connection as an attribute in original training dataset.
To implement the above proposed method (see Figure 1), we design the algorithm showed in Algorithm 1 that incor porates deep learning and knowledge graph for detecting network intrusion. For the experiment, CICIDS2017 dataset is downloaded from "KDD Cup 1999 Data," http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html (accessed on March 12, 2018).

4.2 Preprocessing Section.

As stated by Aggarwal and Sharma [30], CICIDS2017 benchmark intrusion detection dataset is a refined version of CICIDS2017 in which there are 16,000,000 instances in the 10% training dataset. In CICIDS2017 intrusion dataset, four classes of attacks are incorporated, such as remote-to-user (R2L), user-to-root (U2R), denial of service (DoS), and probe in which 22 different attacks are included specifically. In CICIDS2017 dataset, 42 total attributes are identified and incorporated. For the dataset to be suitable for experimentation using machine learning algorithms, the data need to undergo data preprocessing step, where data cleaning, data size balancing, data size reduction, and dimensionality reduction (feature reduction) are performed.

Moreover, sampling and feature selection techniques are applied on the CICIDS 2017 intrusion dataset to produce manageable CICIDS 2017 dataset appropriate for the experiment. Finally, based on the aforementioned activities such as sampling methods, a total of 52,127 instances are prepared for the experiment
Loss: The mean squared error loss function is used. Optimizer: The Adaptive (ADAM) algorithm is selected.

ADAM is the state-of-the-art optimizer for deep neural networks (Schneider et al, 2019).

Betas: These are ADAM optimizer coefficients used for computing running averages of gradient and its square (0.5, 0.999). Learning rate (0.0002).

### 4.3. Attribute Selection.

In building high performance intrusion detection systems, one of the significant research difficulties is effective instances selection from intrusion detection datasets. Accuracy of intrusion detection model has been greatly affected by the presence of irrelevant and redundant attributes in the intrusion detection dataset. As described by Lee et al. [8], 41 attributes were constructed for each network connection on NSL-KDD intrusion detection dataset. To filter best attributes used in constructing DDoS attack detection model to identify abnormal network connections from a given dataset, attribute selection methods have been applied
features present in the training data. That is, for instance for the CICDDoS2019, the number of hidden and visible units for the RBM is 77.

### 4.4. Classification Modeling.

According to Neethu [16], constructing classification model is one of the main challenges for intrusion detection system, which is to construct effective models to identify normal behaviors from abnormal behaviors of network connection by observing collected audit data. In addition, one of the main challenges in intrusion detection systems is learning from static intrusion data to construct a classifier.

Thus, if for instance the autoencoder is trained on a dataset comprising only of benign packets, whenever a benign packet is presented to the autoencoder, we expect that the constructed output should be quite similar and therefore the reconstruction error should be low. However, if this same model is presented with a suspicious packet that is fairly different from the features of benign packet, then we should expect the reconstruction error to be quite high. The same logic can be applied to the restricted Boltzmann machine

With this formulation established, it is easier to frame the classification problem using the autoencoder and RBM. Where in our example, a low reconstruction error means the packet is benign, while a high reconstruction error means the packet is suspicious. Using these predictions,

The NMI provides a means of evaluating the clustering performance of the algorithm by comparing the correlation between the predicted class and the target class. If the predicted and target data are represented as two separate distributions, then we can also apply the NMI to determine performance of non-clustering algorithms.

In this session, we present the experimental results for each model across all datasets. The results are presented in subsections, with each subsection dedicated to a model. For the Autoencoder and Restricted Boltzmann Machine, their subsections consist of plots showing the training and test loss, a table summarizing the performance across the datasets and a detailed discussion of the results. For the rest of the models, they do not optimize a loss function and so only the summary tables and a detailed discussion of the results were presented. Performance evaluations are also carried out using the accuracy and Normalized Mutual score. The innovation of this work lies in the exact detection of the anomality behaviour of the nodes. DDoS attacker tried to affect network in its different forms. The basic nature of DDoS attacker is to flood the network with a large number of packets and then exhaust the network.

The efficiency of planned attack detection system is also assessed against five existing works of [Bellingerite al., (2020)], [Almsgiving et al., (2017)], [Yan Naing Soe,2020] (Naeem Firdous Syed,2020) and [Ashrafi et al., (2013)] who have addressed intrusion detection against DDoS attack using KDD dataset. compares the detection accuracy of the proposed work against the existing works. Recently, [Vellinga et al., (2020)] had proposed Taylor Elephant Herd Optimisation based Deep Belief Network to detect DDoS attacks and attained a classification accuracy of only 83%. Further, [Almsgiving et al., (2017)] had realised Random forest classifier to attain a detection accuracy of 93.77% while [Ashrafi et al., (2013)] work comprising of extended Classifier System with Artificial Neural Network (ANN-XCS) demonstrated a detection accuracy of 98.1% against the proposed work that have implemented the optimization techniques with unsupervised machine learning to achieve a detection accuracy of 99.93%.
This paper focusses on detecting DDoS attack in IoT networks by classifying incoming network packets on the transport layer as either "Suspicious" or "Benign" using unsupervised machine learning algorithms. In this work, two deep learning algorithms and two clustering algorithms were independently trained for mitigating DDoS attacks. We lay emphasis on exploitation based DDOS attacks which include TCP SYN-Flood attacks and UDP-Lag attacks. We use Mirai, BASHLITE and CICDDoS2019 dataset in training the algorithms during the experimentation phase.
Figure 4 shows the desired behavior of the backpropagation training algorithm where the training and validation loss decrease steadily and in unison as the training epoch increases. It is important to point out that the autoencoder is trained to reconstruct SYN-Flood data, meaning it should be unable to reconstruct benign data. We chose the SYN-Flood data for training because there were more instances than the benign data. The same choice is

made for the UDP-Autoencoder model, where we train it on the UDP-Lag data instead of on benign UDP data.

Our experiments show that the random forest (RF) gives better accuracy for normal, DoS, probe, and R2L classes compared to SMO and Bayes Net and it gives the worst accuracy for detecting U2R class of attacks. For U2R class, both SMO and Bayes Net methods give the same perfor mance. There is only a small difference in the accuracy for

```
Input: original training dataset D
Output: classification instance as attack or normal
Use features selection and extract best features
Train machine learning algorithms ML, where ML is machine learning
Select best classifiers such as random forest (RF)
Incorporate RF with D as KB, where KB is the knowledge base
While (new instance == true)
  {
    Apply classifier RF,
    Get class of instance I, as attack or normal, where I is the classified instance
      For (I == true)
      {
        KB fetch classified instance I, where KB consists ML and D
        string comp=compare I with D
        If (comp is not true)
          {
            new instance is not added to D, where D is training dataset
            training dataset not updated
            }
      Else
              {
            new instance is not added to D, where D is training dataset
            training dataset is updated and ready for next training
            New pattern P is generated
            Applied for next classification
          }
      }
  }
```

ALGORITHM 1: Expandable and flexible deep learning method for DDoS attack detection.

Currently, various deep learning algorithms have become very public and concerned more and more benefits in current ages for classifying network connections into normal and abnormal [16].

Some of the popular machine learning algorithms used for classifying a given intrusion audit data include decision tree, support vector machine, neural net- work, genetic algorithm, Naïve Bayesian, and Fuzzy logic. Since the attackers and behaviour of network attacks are be- coming complicated and continuously changing their way of attacking and patterns, it is very difficult to detect several new attacks that come through the network. Therefore, Neethu [16] acclaims that machine learning algorithms applied indifferent intrusion detection researches need an improvement in their classification accuracy.

DDoS modules for SMO and Bayes Net but there is a substantial change for probe modules. Meanwhile U2R and R2L modules have minor training data associated to other modules, it seems that SMO and Bayes Net classifiers give good accuracy with small training datasets. R2L class for RF is better for the RF compared to both SMO and Bayes Net

| SNO | Attributes | Data type | Description |
|---|---|---|---|
| 1 | failed login | Continuous | failed login |
| 2 | logged_ | Discrete | 1 if successfully |
| 3 | Urgent | Continuous | Number of |
| 4 | stabiles | Continuous | No. of data bytes |
| 5 | root shell | Discrete | is received, |
| 6 | errorfree | Continuous | % of connection |
| 7 | srv_serror_rate | Continuous | % of same |
| 8 | same_srv_rate | Continuous | % of connection |
| 9 | errorfree | Continuous | % of connection |
| 10 | protocol type | Discrete | Type of protocol |
| 11 | laceration | Continuous | No. of file |
| 12 | srv_diff_host_rat | Continuous | % of con to diff. |
| 13 | protocol type | continuous | Type of protocol |
| 14 | rv_diff_host_rate | Continuous | % of con to diff. |
| 15 | srv_diff_host_rat | Continuous | % of |
| 16 | protocol type | continuous | Type of protocol |
| 17 | num_file_creations | Continuous | No. of |
| 18 | wrong_fragment | Continuous | No. of wrong |
| 19 | is_host_login | Discrete | 1 if host is |
| 20 | wrong_fragment | Continuous | No. of wrong |
| 21 | is_host_login | Discrete | 1 if host is logged in, 0 otherwise |

As obvious from Tables 2 and 3, all the classifiers con sidered so far could not perform well for detecting all the attacks. To take advantage of the performance of the three classifiers, a random forest (RF) is selected for next inte gration with knowledge base to come up with a scalable and adaptive learning approach for intrusion detection

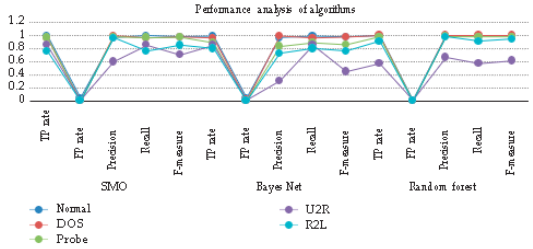
## 5. Discussion of Result

This study investigates that approach Stacked Auto Encoder (SAE) and Con- volitional Neural Network (CNN). for DDoS detection is probable over mixture of Stacked Auto Encoder (SAE) and Con- volitional Neural Network (CNN). As far as my knowledge, this study is the first study which gives practical demo on the likelihoods of hybrid approach for refining DDoS detection. the average accuracy of the three algorithms across the datasets is shown in Figure 2. Firstly, as presented in Figure 2, SMO, Bayes Net, and random forest classifiers have the best average accuracy, i.e., 99.35%, 98.68%, and 99.71%, respectively, when using supervised learning.

Table 1: Performance comparison of algorithms with different evaluation metrics

| Performance metrics | SMO (%) | Bayes Net (%) | Random forest (%) |
|---|---|---|---|
| TP rate | 99.1 | 99.6 | 99.9 |
| FP rate | 1.9 | 2.1 | 0.1 |
| Precision | 99.7 | 98.3 | 99.9 |
| Recall | 99.5 | 98.2 | 99.9 |
| F-measure | 95.4 | 99.4 | 99.9 |
| Accuracy | 99.7 | 98.6 | 99.97 |

Table 2: Performance comparison of algorithms with different evaluation metrics.




| Attack type | SMO (%) | Bayes Net (%) | Random forest (%) |
|---|---|---|---|
| Normal | 99.98 | 99.99 | 99.98 |
| DoS | 97.6 | 97.6 | 99.99 |
| Probe | 97.9 | 89.73 | 99.2 |
| U2R | 88.9 | 86.5 | 86.21 |
| R2L | 77.98 | 80.5 | 92.63 |

Table 3: Performance comparison of the three classifiers.

| | Predicted classes | | | | | |
|---|---|---|---|---|---|---|
| | Normal | DoS | Probe | U2R | R2L | – |
| | 234567 | 4 | 5 | 2 | 3 | Normal |
| Actual classes | ' | 184 | 2 | 0 | 0 | DoS |
| | 4 | 4 | | 0 | 0 | Probe |
| | 4 | 0 | 0 | 5 | 0 | U2R |
| | 9 | 0 | 0 | 3 | 96 | R2L |

The additional problem we confronted in this effort is inaccessibility of prompt data to test the method. So, the method is verified on offline data openly obtainable online. Mostly, the study empirically proofs the option of joining deep learning- and knowledge-graph system for the sake of developing expandable and flexible deep learning method for DDoS attack detection at the same time. We observed that deep learning- and knowledge-graph systems are es- sential to each other. So, our experiment result shows that after integration of machine learning and knowledge base, 99.89% classification accuracy is achieved on the pre- processed NSL-KDD intrusion dataset.

## 6. CONCLUSIONS

This paper presents a novel approach for DDoS attack Detection based on hybrid modular of Stacked Auto Encoder (SAE) and Con- volitional Neural Network (CNN). A hybrid approach of Stacked Auto Encoder with Con- volitional Neural Network (CNN) is proposed for DDoS detection. It is analytically verified with CIC-DDoS2019 dataset of 41,749 entire occurrences, by using ten-fold cross validation. Experimental outcome displays that 99.97% performance is recorded after connection. Once proper formulations are established, the accuracy score can then be used to evaluate both models fairly. Although the autoencoder model is clearly the superior model, the DDOS-Detection class we developed provides methods that allow one to perform network packet classification using either the autoencoder model or the Expectation-Maximization model. The simulation results show that the DDOS-Detection tool built around these models can achieve a net accuracy of as high as 99.71%. Future studies should aim to replicate results in a larger system to detect compromised end-points and also ensure that algorithms are current by possible retraining approaches to handle abnormalities in network performance.

### Data Availability

The dataset used in this work is publicly available as a benchmark for research purposes, https://www.unb.ca/cic/datasets/nsl.html. So, the preprocessed data obtained to support the findings of this work are available from the authors upon request. All the supporting open-source codes for integration activities are available to the research community under an open-source license for the researcher

### Acknowledgment

The authors would like to express their cordial thanks to Dr. Mitsuo Ohta and Dr. Million Meshesha for their valuable advice.

### Conflicts of Interest

The authors hereby declare that there are no conflicts of interest regarding the publication of this paper.

## References

[1] Shirazi, "Evaluation of anomaly detection techniques for scada communication resilience," *IEEE Resilience Week,* 2016.

[2] N. Mirai, "mirai-botnet," 2016. [Online]. Available: https://www.cyber.nj.gov/threat-profiles/botnetvariants/mirai-botnet. [Accessed 31 December 2019].

[3] H. Zhou, B. Liu and D. Wang, "Design and research of urban intelligent transportation system based on the Internet of Things," *Internet of Things,* pp. 572-580, 2012.

[4] S. Lim, S. Yang and Y. Kim, "Controller scheduling for continued SDN operation under DDoS attacks," *Electronic Letter,* pp. 1259-1261, 2015.

[5] A. Buck and E. Govan, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials,* vol. 18.2, 2016.

[6] P. Baldi, "Autoencoders, Unsupervised Learning, and Deep Architectures," *Proceedings of ICML works hop nuns supervised transfer learning,* 2012.

[7] R. Doshi, N. Althorp and N. Feemster, "Machine Learning DDoS Detection for Consumer Internet of Things Devices," *IEEE Deep Learning and Security Workshop,* 2018.

[8] Q. Yan, F. Yu and Q. Gong, "Software defined networking and Distributed denial of service attacks in cloud computing environments," *IEEE Communications Survey & Tutorial,* no. 18, pp. 602-622, 2016.

[9] N. Z. Bawany, J. A. Shamsi and K. Salah, "DDoS Attack Detection and Mitigation Using SDN," *Arabian Journal for Science & Engineering,* no. 2, pp. 1-19, 2017.

[10] B. Kang and H. Choo, "An SDN-enhanced load-balancing technique in the cloud system[J].," *Journal of Supercomputing,* pp. 1-24, 2016.

[11] O. Saniya and D. M. Choo, "Distributed denial of service (DDOS) resilience in cloud," *Journal of Network & Computer Applications,* pp. 147-165, 2016.

[12] H. Luo, Z. Chen and J. Li, "Preventing Distributed Denial-of-Service Flooding Attacks with Dynamic Path Identifiers[J]," *IEEE Transactions on Information Forensics & Security,* pp. 1801-1815, 2017.

[13] U. Dick and T. Schiffer, "Learning to control a structured-prediction decoder for detection of HTTP-layer DDOS attackers," in *Machine Learning*, 2016, pp. 1-26.

[14] Z. Gao and N. Ansari, "Differentiating Malicious DDoS Attack Traffic from Normal TCP Flows by Proactive Tests[J]," *Communications Letters IEEE,* pp. 793-795, 2006.

[15] K. Briceno, A. Rurality and A. Gurov, "Detecting the Origin of DDoS Attacks in OpenStack Cloud Platform Using Data Mining Techniques[M]// Internet of Things," *Smart Spaces, and Next Generation Networks and Systems,* 2016.

[16] N. Hoque, D. Bhattacharyya and J. Kavita, "Botnet in DDoS Attacks: Trends and Challenges[J]," *IEEE Communications Surveys & Tutorials,* pp. 1-1, 2015.

[17] A. Saeed, R. E. Overbill and T. Ridzik, "Detection of known and unknown DDOS attacks using Artificial Neural Networks," *Neurocomputing,* pp. 385-393, 2016.

[18] S. Rama nauseate, N. Geranin and A. Cents, "Modelling influence of Botnet features on effectiveness of DDoS attacks[J]," *Security & Communication Networks,* pp. 2090-2101, 2015.

[19] C. Barghini, M. J. Kavita, S. Singh and D. K. Bhattacharyya, "Anomaly based DDoS attack detection," *International Journal of Computer Applications,* pp. 35-40, 2015.

[20] A. Aggarwal and A. Gupta, "Survey on data mining and IP traceback technique in DDos attack," *International Journal of Engineering and computer science,* vol. 4(6), pp. 12595-12598, 2015.

[21] G. Naima and M. Hemal Atha, "Effective approach towards intrusion detection system using data mining technique," *Egyptian Informatics Journal,* vol. 15(1), pp. 37-50, 2014.

[22] Y. A. Mahmood, "Autoencoder-based feature learning for cybersecurity applications," *International Joint Conference on Neural Networks (IJCNN),* 2017.

[23] S. Yadav and S. Subramanian, "Detection of Application Layer DDoS attack by feature learning using Stacked Auto Encoder," *International Conference on Computational (ICCTICT),* 2016.

[24] A. Fischer and C. Ige, "An introduction to restricted Boltzmann machines. In Libero American congress on pattern recognition," *Springer, Berlin, Heidelberg,* pp. 14-36, 2012.

[25] V. G. Rydin and G. Volcano, "An expectation maximization method to estimate a rank-based," 2017.

[26] D. Ferrierite, "Extreme Dimensionality Reduction for Network Attack Visualization with Autoencoders," (IJCNN), 2019.

[27] I. Sharfuddin, A. H. Lashkar, S. Haka and A. Ghobadi, "Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy," International caravan conference on security (ICCST). IEEE, pp. 1-8, 2019.

[28] Y. Maidan, M. Bandana, Y. Mathur, Y. Mirsky and Shabtai, "Network based detection of iot botnet attacks using deep autoencoders," IEEE Pervasive Computing, pp. 12-22, 17(3).

[29] C. Elkan, "Using the triangle inequality to accelerate k-means," ICML-03, pp. 147-153, 2003.

[30] R. Bhatia, S. Benno, J. Esteban, T. V. Lakshman and J. Grogan, "Unsupervised machine learning for network-centric anomaly detection in IoT.," in the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks.

Hailye Tekleselase Michael holds a BSc degree in Information Systems from University of Gondar, and MSc degree in Information Systems from Addis Ababa University. His current research interests are: Cyber Security, Big Data, AI, ICT and Mobile Computing. He currently Instructor at Wolaita Sodo University, He is a member of the Ethiopian Space Science Society (ESSS) and the Institution of Electrical and Electronics Engineers (IEEE).