Proceedings of the 2nd International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/67/20240695

Beyond signatures and thresholds: Intelligent DDoS detection using Convolutional Neural Networks

Yunze Li

The Hong Kong Polytechnic University

yunze.li@connect.polyu.hk

Abstract. Distributed Denial of Service (DDoS) attacks have evolved into sophisticated threats against digital infrastructure. This research investigates Convolutional Neural Networks (CNNs) for DDoS detection using the CIC-IDS2017 dataset. A CNN model was developed, leveraging convolutional and pooling layers for hierarchical feature extraction. Preprocessing ensured dataset consistency and generalization. The model achieved 99.978% test accuracy, demonstrating proficiency in recognizing DDoS patterns. Comparatively, a Deep Neural Network (DNN) benchmark obtained 99.9512% accuracy, indicating CNNs' superiority. However, CNN training time was longer. While results highlight deep learning's potential against DDoS attacks, optimizations for real-time deployment warrant exploration. This empirical evaluation and comparative analysis enriches the discourse on utilizing machine learning for robust cybersecurity.

Keywords: DDoS detection, Convolutional neural networks, Machine learning, Deep learning, Cyber security.

1. Introduction

Distributed Denial of Service (DDoS) attacks, once a theoretical cyber threat, have evolved into a commonplace menace, plaguing the vast digital infrastructures of the 21st century. In essence, a DDoS attack is a coordinated attempt to incapacitate online services by inundating them with a deluge of traffic from manifold sources. The outcome is a crippled system, staggering under the sheer volume of requests, leading to either a complete shutdown or a significant decline in service quality [1]. Orchestrated through an army of botnets – a collection of compromised computers – the attacker aims not just at a single system, but casts a net wide enough to paralyze entire networks or servers [2].

The ramifications of DDoS attacks within the intricate web of global digital infrastructure are profound, casting a long shadow over the technological advancements of the 21st century. As contemporary societies gravitate increasingly towards digitization, the reliance on virtual platforms, spanning from finance and healthcare to governance and commerce, has reached staggering proportions. Consequently, a successful DDoS incursion extends beyond the mere disruption of a singular digital service; it poses a tangible risk of impeding the fundamental mechanisms underpinning modern societal operations, and can incapacitate entire sectors, crippling economies and undermining trust in digital technologies [3].

Historically, the bulwark against DDoS attacks has been a combination of intrusion detection systems (IDS), fortified firewalls, meticulous traffic analysis, and rate-based filters [4]. Though these have

DOI: 10.54254/2755-2721/67/20240695

served as formidable defense mechanisms in the past, the ever-evolving nature of DDoS attacks, coupled with their increasing sophistication, has exposed the chinks in their armor. Three primary shortcomings of these conventional methods are:

Reactivity Over Proactivity: Rooted in a signature-based foundation, traditional mechanisms are adept at tackling known threats. However, they falter when confronted with zero-day exploits or newly devised DDoS stratagems [5].

Scalability Concerns: The modern DDoS attacks are not just diverse but also colossal in their volume. Legacy systems, not designed for such an onslaught, often find themselves outpaced, unable to analyze traffic in real-time [3].

The False Positive Dilemma: The propensity of threshold-based techniques to flag legitimate traffic as malicious leads to false positives [6]. This not only disrupts genuine users but also allocates precious resources to chase these false alarms.

Enter the realm of Convolutional Neural Networks (CNNs). Originating primarily from endeavors in the image processing sphere, CNNs gained prominence not merely for their prowess in discerning visual patterns but also for their inherent architecture that facilitates a hierarchical extraction of features from raw data. This architectural marvel has since allowed CNNs to extrapolate their capabilities to diverse arenas, ranging from the nuances of natural language processing, the intricacies of genomics, to the challenges of medical diagnoses. Whether tasked with translating a foreign language, predicting genetic mutations, or diagnosing rare medical conditions, CNNs have consistently demonstrated an uncanny ability for pattern recognition and classification. Their exceptional talent lies in their multilayered structure, wherein each layer refines and builds upon the previous one, making them exceptionally equipped for intricate tasks.

The challenges posed by network traffic, especially during a DDoS scenario, are labyrinthine. Yet, it's here that CNNs, with their finesse in discerning patterns amid chaotic datasets, can truly shine. Their propensity to learn, adapt, and recognize patterns in a sea of data noise makes them potent contenders for detecting established and emerging DDoS threats.

In light of the aforementioned context, this investigation delves into the potential of Convolutional Neural Networks (CNNs) for the detection of DDoS attacks. The primary dataset under scrutiny is the CIC-IDS2017, characterized by its comprehensive representation of network behaviors, serving both as a training and testing benchmark [7]. The motivation behind this study is twofold: initially, it aims to evaluate the efficacy of CNNs in accurately detecting DDoS attacks; subsequently, it seeks to contrast the training duration and resultant accuracy between CNNs and their traditional counterpart, the Deep Neural Network (DNN). Such a comparative analysis underpins the overarching objective of gauging practicality and efficiency within the domain of network security. This paper endeavors to not only elucidate empirical findings but also to contribute meaningfully to the discourse on cybersecurity, advocating for enhanced digital defenses against the pervasive threat of DDoS attacks.

2. Method

2.1. Dataset

This research leverages the capabilities of the CIC-IDS2017 dataset. Curated by the Canadian Institute for Cybersecurity at the University of New Brunswick, the dataset is revered for its authentic representation of DDoS attack traffic, meticulously executed in a controlled network environment. Among its myriad of attack samples, it encompasses prevalent DDoS vectors like the TCP flood, UDP flood, and HTTP flood. Its foundational relevance to the real-world context ensures its significance in constructing pragmatic models that can discern and detect these threats in live settings.

Upon obtaining the raw CSV records from the dataset, an assortment of flow-based features was extracted, including but not limited to source and destination IPs, ports, and flow sizes. The categorical labels, essential for supervised learning, were transformed from their textual representation to numerical entities through the utilization of LabelEncoder.

DOI: 10.54254/2755-2721/67/20240695

The pre-processing of the extracted features was both meticulous and multifaceted. Preliminarily, any data voids or missing values detected were addressed by imputing the median values, ensuring continuity and coherence. Subsequently, features of numerical essence, like flow sizes, were subjected to normalization through the StandardScaler. By transforming these features to exhibit a zero mean and unit variance, the model is insulated from disproportionate feature influence, guaranteeing a stable and coherent learning trajectory.

The dataset then underwent a dimensional transformation, where the processed features were augmented with an additional dimension, rendering them congruent with the Conv1D convolution layers' requirements. Adhering to best practices, the dataset was bifurcated with 80% allocated for model training and the remaining 20% earmarked for evaluation.

This rigorous preprocessing ensures that the CIC-IDS2017 data, rich in its DDoS samples, is harmonized and standardized, priming it for effective consumption by deep learning architectures. Such meticulous preparation not only enables accurate model training but also underlines the importance of data quality and structure in the pursuit of achieving state-of-the-art DDoS detection.

2.2. Model Structure

Before delving into the specifics of the CNN architecture, it is paramount to understand the rationale behind employing this specific type of neural network. Convolutional Neural Networks have garnered substantial acclaim in recent years, especially in applications that involve image or sequence data. The hierarchical pattern in CNNs allows the network to recognize local patterns in early layers and more abstract concepts in deeper layers, making them particularly adept for tasks that require recognition of spatial hierarchies. This section elucidates the layers employed, their functionalities, and their importance in DDoS attack detection.

The CNN employed in this study is predominantly structured as Table 1:

Table 1. Model Structure.

Layer Name	Layer Type	Output Shape	Parameters
conv1d	Conv1D	(None, 77, 32)	96
batch_normalization	BatchNormalization	(None, 77, 32)	128
max_pooling1d	MaxPooling1D	(None, 38, 32)	0
conv1d_1	Conv1D	(None, 37, 64)	4160
batch_normalization_1	BatchNormalization	(None, 37, 64)	256
max_pooling1d_1	MaxPooling1D	(None, 18, 64)	0
flatten	Flatten	(None, 1152)	0
dense	Dense	(None, 128)	147,584
dense_1	Dense	(None, 1)	129
Summary:			
Total params:			152,353
Trainable params:			152,161
Non-trainable params:			192

Each layer's role and configuration have been carefully curated to optimize the detection of DDoS patterns in the data:

- Conv1D Layer: Serving as the initial layer, the Conv1D is tailored for sequences, an apt choice for time-series data or any data type arranged sequentially. With 32 filters and a kernel size of 2, this layer scans the data in chunks of two, detecting patterns and motifs in the data.
- BatchNormalization Layer: Introduced after the Conv1D layer, this layer normalizes the activations of the neurons, ensuring a smoother training process. It adjusts and scales the activations such that they maintain a mean output close to 0 and a standard deviation close to 1.

DOI: 10.54254/2755-2721/67/20240695

- MaxPool1D Layer: Pooling layers contribute to the model's ability to discern and retain essential features while simultaneously reducing the spatial size of the representation, enhancing computational efficiency. With a pool size of 2, this layer essentially down-samples the output from the preceding Conv1D layer by half.
- Dropout Layer: Inculcated within the architecture at regular intervals, these layers assist in preventing overfitting. By randomly setting a fraction (20% in our model) of input units to 0 during training, the model becomes more robust and generalizable.
- Flatten Layer: Transitioning from convolutional layers to dense layers necessitates the reshaping of the data, and this is precisely the role the Flatten layer undertakes.
- Dense Layer: As fully connected layers, the dense layers receive input from all neurons of the preceding layer, ensuring holistic information processing. The penultimate dense layer comprises 128 neurons and employs a Rectified Linear Unit (ReLU) activation function. The final dense layer, streamlined for binary classification, consists of a single neuron with a 'sigmoid' activation function, outputting a probability indicative of potential DDoS activity.

2.3. Hardware Configuration

To ensure reproducibility and clarity regarding the computational resources utilized during the experiments, it's imperative to detail the hardware configuration on which the model was trained. Given that deep learning models, especially convolutional neural networks, can be computationally intensive, the specific hardware setup can have significant implications on training times and overall performance.

Table 2 encapsulates the hardware details of the environment in Google Colab where our experiments were conducted:

Table 2. Hardware Configuration.

Hardware Component	Specification
CPU	
Vendor	GenuineIntel
Model	Intel(R) Xeon(R) CPU @ 2.30GHz
Cores	1 Core
Threads	2 Threads
Frequency	2.30 GHz
Cache Size	46080 KB
Memory (RAM)	
Total Memory	13294252 kB (~13 GB)
Free Memory	10331268 kB (~10.3 GB)
GPU	
Model	Tesla T4
Memory	15360 MiB
Driver Version	525.105.17
CUDA Version	12.0
Operating System	
OS	Ubuntu 22.04.2 LTS
Codename	jammy

The table provides a snapshot of the CPU and GPU specifications, alongside memory details. Such specifications give insights into the computational capabilities leveraged during model training and evaluation, facilitating comparisons and benchmarking with other setups.

DOI: 10.54254/2755-2721/67/20240695

2.4. Model Training

The holistic training of the model remains pivotal to its efficacy. The choice of the Binary Cross-Entropy loss function aligns seamlessly with the binary classification objective of the model [8]. Given the nature of the problem, any deviation between the predicted probability and actual class label is penalized, ensuring model refinement throughout the training process.

The Adam optimizer was employed, distinguished by its adaptive learning rates and momentum. Its efficiency in handling sparse gradients and its capability to adjust the learning rate on-the-fly, based on observed patterns in the training data, made it an optimal choice [9].

Concerning hyperparameters, a preliminary learning rate was set, with provisions for adjustments contingent on validation loss improvements. The batch size was chosen considering the balance between computational efficiency and the granularity of weight updates.

To preclude potential overfitting and achieve an optimal model, an EarlyStopping strategy was implemented. Monitoring the validation loss, this mechanism halts the training once no substantial improvements are discerned for a stipulated number of epochs, ensuring the model retains its generalization capabilities without veering towards overfitting [10].

2.5. Benchmark with DNN

To comprehensively evaluate the capabilities of the proposed CNN model for DDoS detection, a DNN model was adopted as a comparative benchmark (see Table 3). The ubiquitous adoption of DNNs across myriad domains underlines its versatility and efficacy as a learning model [11].

The simplicity and strong representational power of DNNs render it an ideal choice as a normalized performance benchmark. Comparing against such a canonical model provides contextual insights into any improvements observed with the proposed CNN. The DNN model was trained in a similar fashion as the CNN, utilizing binary cross-entropy loss, Adam optimizer, and regularization to ensure a fair comparison.

Leveraging such a widely-used model and comparing it against the CNN strengthens the evaluation process and provides a normalized backdrop for assessing the capabilities of convolutional networks for DDoS detection.

Table 3. DNN Model Structure Parameters

Layer (type)	Output Shape	Parameters
dense (Dense)	(None, 128)	10,112
batch_normalization (BatchNormalization)	(None, 128)	512
dense_1 (Dense)	(None, 64)	8,256
batch_normalization_1 (BatchNormalization)	(None, 64)	256
dense_2 (Dense)	(None, 32)	2,080
batch_normalization_2 (BatchNormalization)	(None, 32)	128
dense_3 (Dense)	(None, 1)	33
Summary:		
Total params		21,377
Trainable params		20,929
Non-trainable params		448

3. Results and Discussion

Following the systematic construction and comprehensive training of the deep learning models, their performance was critically evaluated. The efficacy was determined by the models' ability to precisely classify DDoS attack patterns within the CIC-IDS2017 dataset.

DOI: 10.54254/2755-2721/67/20240695

3.1. Performance on the Test Set

Convolutional Neural Network (CNN): The CNN, when evaluated on the test set, exhibited exceptional accuracy, reflecting its robustness and precision. The model recorded a test accuracy of approximately 99.978%, emphasizing its adeptness at detecting DDoS patterns.

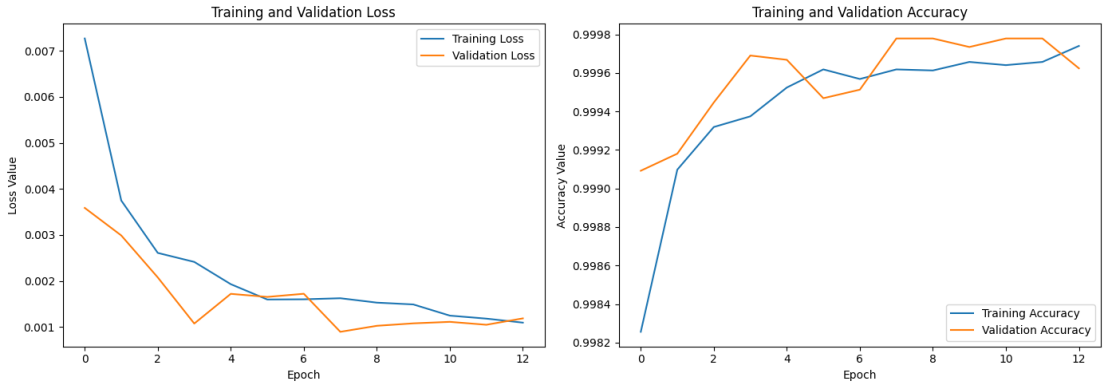

Figure 1. Training and evaluation metrics for the CNN model.

Figure 1 illustrates the CNN's training trajectory. The model's loss was observed to be 8.9504e-04, indicating swift convergence towards an optimal solution.The CNN model required 481 seconds for the training process.

Deep Neural Network (DNN): In comparison, the DNN, while demonstrating commendable performance, was slightly outperformed by the CNN. The DNN achieved a test accuracy of 99.9512%.The DNN model's training was completed in 438 seconds.

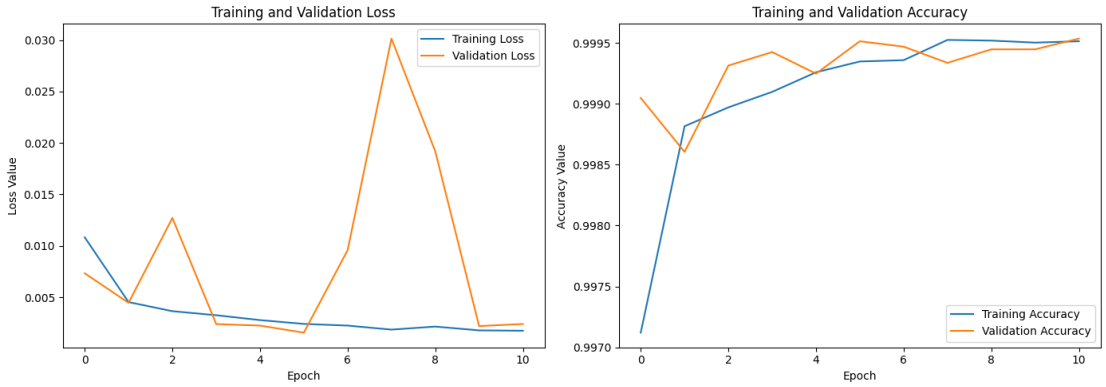

Figure 2. Training and evaluation metrics for the DNN model.

As shown in Figure 2, it was observed that the loss curve for the DNN manifested greater volatility in comparison to the CNN. Such variations might suggest the model's heightened sensitivity to dataset intricacies, which could be a factor contributing to its marginally reduced accuracy.

3.2. Comparative Analysis with Traditional Methods

The performance of the deep learning models, although impressive, was contextualized by comparing their outcomes with established traditional methods. Classical methodologies, ranging from decision trees to support vector machines, have been historically utilized for DDoS detection. The near-perfect accuracy of the CNN, combined with its stable loss progression, indicates its potential superiority over these conventional algorithms. Nevertheless, a systematic benchmarking against each traditional approach is recommended for a comprehensive evaluation.

DOI: 10.54254/2755-2721/67/20240695

3.3. Interpretation of Results and Their Implications

The results derived from the CNN and DNN models possess significant implications in the domain of cybersecurity. These high accuracy rates underscore the potential of deep learning architectures in such contexts. Furthermore, the findings suggest that diligent preprocessing and astute feature extraction from the CIC-IDS2017 dataset lead to robust models capable of real-world DDoS detection.

Additionally, contrasting the performances of the CNN and DNN offers a salient insight. While both architectures have exhibited potency, the inherent spatial feature extraction capabilities of the CNN seem to provide it with an advantage in tasks that demand recognition of sequential patterns, such as DDoS detection.

3.4. Limitations and Recommendations for Future Research

Despite the promising results, several limitations warrant acknowledgment. Firstly, there is the potential susceptibility to overfitting, especially visible in the DNN's loss curve, despite the inclusion of dropout layers. Secondly, the scope of the CIC-IDS2017 dataset, while comprehensive, might limit the models' generalizability to diverse or evolving attack vectors. Lastly, when considering the deployment in real-time scenarios, the CNN's longer training duration and increased computational demands compared to the DNN could be constraining factors.

The more substantial computational requirement of the CNN implies higher resource costs, which might not be feasible for all deployment environments. Additionally, while the CNN outperforms the DNN in terms of accuracy, its longer training time could be a significant concern in dynamic scenarios where models need frequent retraining.

For future exploration, it would be beneficial to study the models' efficiency in real-time settings, where detection has to be both accurate and immediate. Investigating potential optimizations for the CNN to reduce its training time without compromising its accuracy would be instrumental. Additionally, ensemble techniques, which leverage the strengths of various architectures, could offer a balanced solution in terms of accuracy, training time, and computational cost.

4. Conclusion

This investigation assessed the proficiency of Convolutional Neural Networks (CNNs) in discerning Distributed Denial of Service (DDoS) threats within the context of the CIC-IDS2017 dataset. A parallel objective was to juxtapose the performance of CNNs with Deep Neural Networks (DNNs), delineating their respective efficacies in cybersecurity applications.

Empirical findings underscore the acumen of CNNs in pinpointing DDoS attack nuances, registering an accuracy exceeding 99% during tests. The strategic exploitation of temporal features by CNNs, augmented by rigorous data preprocessing, fortified its ability to learn from the dataset's myriad samples. By comparison, the DNN, though commendable in its right, recorded a slightly subdued accuracy, signifying the inherent edge CNNs possess in handling sequential data tasks.

Yet, there are pragmatic considerations that emerge from this analysis. The extended training duration of CNNs translates to heightened computational demands. Such computational intensities, despite yielding superior accuracy, may challenge their integration into latency-stringent scenarios. The observance of possible overfitting tendencies further necessitates exploration into hybrid techniques, potentially integrating multiple model architectures.

In essence, the study accentuates the potential of deep learning, especially CNNs, in detecting intricacies within network traffic. With DDoS attacks continually metamorphosing in complexity, there's an evident shift from traditional signature-based defenses to more agile, data-driven approaches. This investigation not only validates the efficacy of CNNs but also paves the way for translating such theoretical insights into tangible security measures. Subsequent research endeavors could delve into real-time detection enhancements and strategies to counteract nascent attack modalities.

DOI: 10.54254/2755-2721/67/20240695

References

[1] Gupta, B.B.,, Misra, M., and Joshi, R.C., 2008. An ISP level solution to combat ddos attacks using combined statistical based approach. *International Journal of Information Assurance and Security*, 3(2), pp.102–110.

[2] Antonakakis, M.,, April, T.,, Bernhard, M.,, Bursztein, E.,, Cochran, J.,, Halderman, A.,, Invernizzi, L.,, Kallitsis, M.,, Kumar, D.,, Lever, C.,, Halderman, J.A.,, Ma, Z.,, Mason, J.,, Menscher, D.,, Seaman, C.,, Sullivan, N.,, Thomas, K., and Zhou, Y.Z., 2017. *Understanding the Mirai botnet*. [online] Understanding the Mirai Botnet. Available from: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis [Accessed 28 Aug. 2023].

[3] Mirkovic, J., and Reiher, P., 2004. A taxonomy of ddos attack and ddos defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34(2), pp.39–53.

[4] Wang, B.,, Zheng, Y.,, Lou, W., and Hou, Y.T., 2014. DDoS attack protection in the era of cloud computing and software-defined networking. *2014 IEEE 22nd International Conference on Network Protocols*, pp.624–629.

[5] Wang, H.,, Zhang, D., and Kang , G.S., 2002. Detecting syn flooding attacks. *Proceedings.Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, pp.1530–1539.

[6] Saied, A.,, Overill, R.E., and Radzik, T., 2016. Detection of known and unknown ddos attacks using artificial neural networks. *Neurocomputing*, 172, pp.385–393.

[7] Sharafaldin, I.,, Habibi Lashkari, A., and Ghorbani, A.A., 2018. Toward generating a new intrusion detection dataset and intrusion traffic characterization. *Proceedings of the 4th International Conference on Information Systems Security and Privacy*.

[8] Usha, R., Vamsidhar, Y., 2020. Binary cross entropy with deep learning technique for Image classification. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(4).

[9] Kingma, D.P., and Ba, J., 2017. Adam: A method for stochastic optimization. [online] arXiv.org. Available from: https://arxiv.org/abs/1412.6980 [Accessed 28 Aug. 2023].

[10] Yingbin, B., Bo, H., Yanhua, Y., Jiatong, L., Yinian, M., Guang, N. Tongliang, L., 2021. Understanding and Improving Early Stopping for Learning with Noisy Labels. *35th Conference on Neural Information Processing Systems.*

[11] Madushi, H. P., Yogachandran, R., Safak, D., Ahmet, M. K., and Rongxing, L., 2022. Deep Learning for Encrypted Traffic Classification and Unknown Data Detection [online] arXiv.org. Available from: https://arxiv.org/abs/2203.15501 [Accessed 28 Aug. 2023].