Computers & Security 144 (2024) 103962

Contents lists available at ScienceDirect

Computers & Security

journal homepage: www.elsevier.com/locate/cose

Robust DDoS attack detection with adaptive transfer learning

Mulualem Bitew Anley, Angelo Genovese [*], Davide Agostinello, Vincenzo Piuri

Department of Computer Science, Università degli Studi di Milano, Italy

ARTICLE INFO

Keywords:
DDoS
Cyber security
Deep learning
Transfer learning

ABSTRACT

In the evolving cybersecurity landscape, the rising frequency of Distributed Denial of Service (DDoS) attacks requires robust defense mechanisms to safeguard network infrastructure availability and integrity. Deep Learning (DL) models have emerged as a promising approach for DDoS attack detection and mitigation due to their capability of automatically learning feature representations and distinguishing complex patterns within network traffic data. However, the effectiveness of DL models in protecting against evolving attacks depends also on the design of adaptive architectures, through the combination of appropriate models, quality data, and thorough hyperparameter optimizations, which are scarcely performed in the literature. Also, within adaptive architectures for DDoS detection, no method has yet addressed how to transfer knowledge between different datasets to improve classification accuracy. In this paper, we propose an innovative approach for DDoS detection by leveraging Convolutional Neural Networks (CNN), adaptive architectures, and transfer learning techniques. Experimental results on publicly available datasets show that the proposed adaptive transfer learning method effectively identifies benign and malicious activities and specific attack categories.

1. Introduction

Distributed Denial of Service (DDoS) attacks are a significant threat to organizations worldwide (Chadd, 2018). These attacks have the potential to paralyze networks, making them inaccessible to legitimate users and causing severe disruptions in service availability and integrity. The ability to detect and mitigate DDoS attacks has therefore become vital to ensuring the resilience and security of critical infrastructure. Given the rising frequency and complexity of DDoS attacks in the cybersecurity landscape, it is imperative to develop effective intrusion detection systems (IDS) to ensure network infrastructure integrity and availability. Deep learning (DL) models have emerged as a promising approach for detecting and mitigating such attacks by automatically learning complex patterns from network traffic data (Diro and Chilamkurti, 2018; Gümüşbaş et al., 2020) and various DL models are being developed to enhance the detection of DDoS attacks. However, mainly due to the dynamic nature of attackers' behavior and evolving cyber threats, maintaining up-to-date models can be a challenging task (Kolias et al., 2017). Furthermore, developing DL models for intrusion detection faces another significant challenge due to the limited availability of data required for effective training, with the consequence that the scarcity of adequately sized and high-quality training datasets hinders the widespread adoption of DL in IDSs. To mitigate this aspect, transfer learning approaches have been considered to train DL models by leveraging data originating from different sources and increasing detection accuracy (Das et al., 2022). However, no method in the literature has yet considered the design of DL models trained using transfer learning that can adapt to evolving attacks by using adaptive architectures.

This paper proposes a novel methodology based on DL for DDoS detection that leverages adaptive architectures in a transfer learning modality, to achieve an accurate classification of benign vs. malicious networks in evolving scenarios. Our approach employs customized CNN models with diverse layer configurations, in addition to several publicly available models such as VGG16, VGG19, and ResNet50. We train the models, considering both a binary and a multi-label classification, by adopting transfer learning techniques while adaptively optimizing hyperparameters, introducing a dynamic and flexible approach that enhances the robustness and efficiency of DDoS attack detection.

The remainder of the paper is structured as follows. Section 2 provides an overview of related works in the field of DL and transfer learning-based DDoS detection and hyperparameter tuning. Section 3 presents the methodology and framework employed in our proposed approach. Section 4 discusses the results and performance analysis of our proposed methodology. Finally, Section 5 concludes the paper.

2. Related works

In the context of DDoS attack detection, various studies have employed DL techniques with significant success. The papers by Sabeel

[*] Corresponding author.
E-mail addresses: mulualem.anley@unimi.it (M.B. Anley), angelo.genovese@unimi.it (A. Genovese), davideagostinello@gmail.com (D. Agostinello), vincenzo.piuri@unimi.it (V. Piuri).

https://doi.org/10.1016/j.cose.2024.103962
Received 6 March 2024; Received in revised form 21 May 2024; Accepted 18 June 2024
Available online 22 June 2024

et al. (2019) and Cil et al. (2021) evaluate the effectiveness of DL techniques to improve the detection accuracy of DDoS attacks. The work presented in Shaaban et al. (2019) delves into the application of the CNN models for large-scale DDoS attack detection within software-defined networks (SDN). Furthermore, Chen et al. (2019) and Nugraha and Murthy (2020) introduced multi-channel CNN and hybrid CNN-LSTM (Long Short-Term Memory) models. Another study (Yeom et al., 2022) introduced a collaborative LSTM-based DDoS detection framework to address the challenges of irregular traffic patterns. These studies showed the promising potential of CNN-based DL models for the efficient detection of DDoS attacks.

Differently from the approaches presented above, which only consider a single DL model, the method described in Elsaeidy et al. (2021) combines the strengths of various models to enhance both the accuracy and the robustness of detection systems. Furthermore, Wei et al. (2021) demonstrated the effectiveness of integrating a Multilayer Perceptron (MLP) with an Autoencoder (AE) for DDoS detection and classification. Complementing these advancements, Hnamte and Hussain (2023) proposes a hybrid model combining CNNs and Bidirectional Long Short-Term Memory (BiLSTM) networks. This approach leverages CNNs' ability in feature extraction and pattern recognition, alongside BiLSTMs' capability to understand sequence and temporal dependencies in the data streams.

In exploring the complex landscape of DDoS attacks, it is essential to recognize the heterogeneity of these threats and gain a comprehensive understanding of the advanced defensive mechanisms required for protecting cloud-based infrastructures. To this purpose, the work in Agrawal and Tapaswi (2019) highlighted the various DDoS attacks and their corresponding defensive approaches to protect cloud infrastructures. Moreover, the work in Venkatesan et al. (2016) presented a moving target defense technique, shifting proxy servers and remapping client connections, effectively disrupting attackers' efforts to map out and exploit network vulnerabilities. Similarly, Kansal and Dave (2017) introduced a method that uses load-balancing algorithms alongside attack proxies to differentiate between malicious insiders and genuine clients, adding an extra layer of security. Furthermore, Jia et al. (2014) developed a cloud-enabled defense mechanism that employs selective server replication and intelligent client reassignment, effectively turning victim servers into dynamic targets to isolate attacks.

In response to the prevalent challenges of scarce labeled data in developing DL models for DDoS detection, current research emphasizes the integration of transfer learning techniques. This kind of approach leverages knowledge from pre-trained models, which have been trained on extensive datasets, to enhance learning efficiency and accuracy in tasks constrained by limited labeled data availability (Masum and Shahriar, 2021). Such as the method described in Wu et al. (2019), which demonstrates the effectiveness of transfer learning in IDS, leveraging knowledge from pre-trained models. Transfer learning has also been applied for DDoS attack detection in IoT environments. For example, the work by Okey et al. (2023), Zhang et al. (2021), Rodríguez et al. (2022), Xue et al. (2022) and Vu et al. (2020) has demonstrated the adaptation of pre-trained DL models for IDS in IoT. Furthermore, the works presented in Yang and Shami (2022) proposed a CNN-based transfer learning approach specifically tailored for IDS in the Internet of Vehicles (IoV).

Although DL models have demonstrated proficiency in identifying known cyber threats, they often face challenges in detecting new or evolving DDoS attack patterns. To address this challenge, adaptive DL techniques have been proposed for DDoS attack detection. As an example, the work described in Cheng et al. (2018) introduced a method based on multiple-kernel learning, while Kushwah and Ranga (2021) employed an improved self-adaptive evolutionary extreme learning approach. Furthermore, the method introduced in Agostinello et al. (2023) consists of a DL approach for DDoS attack detection using adaptive architectures with an optimized number of neurons.

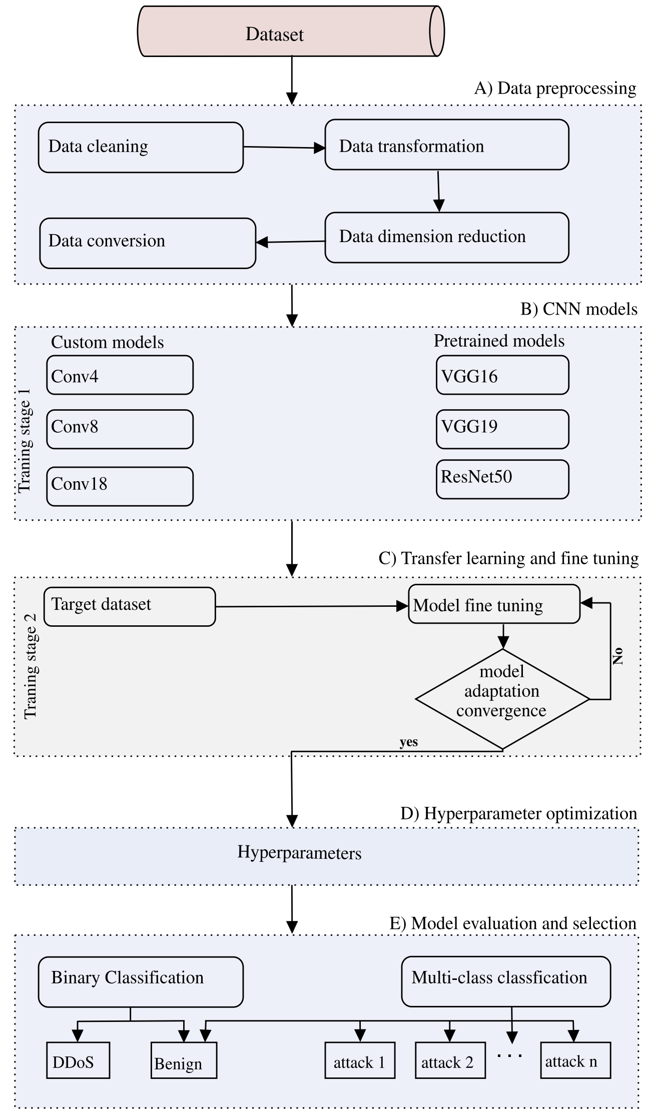

Fig. 1. A comprehensive methodology framework for robust transfer learning DDoS attack detection, encompassing *(A)* data preprocessing, *(B)* CNN model, *(C)* transfer learning and fine-tuning, *(D)* hyperparameter optimization, and *(E)* model evaluation and selection.

While DL-based approaches for DDoS detection using transfer learning or adaptive architectures have been proposed in the literature, to the best of our knowledge, no approach has yet considered adaptive architectures in a transfer learning modality. To address these gaps, our paper proposes an adaptive DL approach for DDoS detection within a transfer learning framework.

3. Methodology

This section explains our proposed framework for DDoS detection using DL models trained using the adaptive transfer learning procedure. The methodology comprises five steps: (*i*) data preprocessing, (*ii*) CNN models, (*iii*) transfer learning, (*iv*) hyper parameter optimization, and (*v*) model evaluation and selection. Fig. 1 outlines the proposed methodological framework.

3.1. Data preprocessing

Data preprocessing consists of (*i*) data cleaning, (*ii*) data transformation, (*iii*) data dimensionality reduction, and (*iv*) data conversion

Data cleaning. We initially focused on validating and correcting inconsistencies and errors within the dataset to ensure its integrity for model training. First, we removed columns lacking useful values, including socket-related features, and those filled solely with zeros. We then eliminated duplicate rows and rows containing NaN values. Finally, we replaced all infinite and null values with −1.

Data transformation. This task encompasses dataset transformation aimed at ensuring consistent numerical values across diverse datasets. Initially, we achieve this by normalizing numerical values within the [0, 1] range through the min–max method. Additionally, categorical features undergo label encoding, which converts categorical values into numerical counterparts. This process utilizes two methods: a label encoder, which converts each label into a unique numerical value, and a one-hot encoder (OHE), which transforms labels into n-dimensional binary vectors, where n is the number of labels.

Data dimensionality reduction. This step aims to reduce the number of features to decrease noise, accelerate training, and achieve a consistent number of features across diverse datasets. To execute this reduction, we apply the PCA technique by determining the optimal number of principal components using the maximum likelihood estimation (MLE) method, a statistical approach for estimating the parameters of a probability distribution that best describes a set of observed data (Ogbuanya, 2021).

Data conversion. The pre-trained CNN models that we consider in this paper, VGG16, VGG19, and ResNet50, have been trained on image datasets. However, network traffic datasets are typically captured in non-image formats, such as .csv or .pcap formats. To enhance the effectiveness of DDoS attack detection through the application of transfer learning, it is important to transform this non-image network traffic data into an image-compatible format suitable for CNNs.

We first scale the numeric features of each dataset to a range of [0, 1] to normalize the data. Following this initial normalization, we apply the quantile transform technique to each feature. This method involves discretizing the normalized values into quantiles, which are then mapped onto a new scale ranging from 0 to 255. This adjustment aligns the data values with the standard range of pixel intensities used in image processing, facilitating their interpretation as image pixels. Using this quantile-scaled data, we generate images for each category within the datasets, including various types of network attacks and benign traffic.

Initially, these images are created with dimensions of 9 × 9 pixels and are encoded in three color channels (RGB), which allows us to capture and distinguish a broad spectrum of feature variations through color differentiation. If the number of features is lower, we add padding to maintain consistency. To ensure that these images are compatible with commonly used pre-trained models such as VGG16, VGG19, and ResNet50, we standardize the dimensions of these initial 9 × 9 images to 224 × 224 pixels, maintaining a three-channel (RGB) format.

3.2. CNN models

In our work, we consider three different customized CNN DL architectures to evaluate the behavior under CNNs with varying depths for one-dimensional input vectors, namely (*i*) Conv4, (*ii*) Conv8, and (*iii*) Conv18 and three pre-trained models, specifically VGG16, VGG19 and ResNet50. For each architecture, we explore two variants of classification types: one conducts binary classification, distinguishing benign from DDoS attacks, and the other performs multi-label classification, aiding in the identification of each specific type of attack. Below, we elaborate on the configurations of these customized CNN architectures.

Conv4. The customized four-layer CNN applies convolutional processing to the input data, enhances the model's non-linearity with ReLU activation functions after the first and third convolutional layers, and utilizes max-pooling operations to down-sample the data for improved feature extraction.

In this paper, a 1D CNN architecture with 4 layers is designed to meet our task's demands. Illustrated in Fig. 2, the model begins with an input layer $(N, 1)$, followed by Conv1D operations $(Conv_i)$ using F filters, K-sized kernels, and *relu* activation. After the convolution operation, global average pooling actively reduces the spatial dimensions. To prevent overfitting and improve generalization, dropout and regularization techniques (L1/L2) are incorporated into the architecture. Dropout layers with rates between 0.1 and 0.5 are inserted after each Conv1D layer, and regularization is applied to the convolutional layers. A dense layer with output dimension H and *relu* activation is next. The final layer is a dense layer with O output classes and *softmax* activation.

Conv8. Building upon Conv4, we extend our model with 8 convolution layers. This expansion enables us to capture more complex and abstract patterns within the data. The architecture depicted in Fig. 2 includes added layers that facilitate a deeper feature extraction process, empowering the model to excel in tasks that demand a higher level of complexity and feature representation.

Conv18. We extend Conv4 and Conv8 by incorporating 18 convolution layers. This model, with its increased depth, captures an even larger hierarchy of features in the dataset representations (see Fig. 2).

3.3. Transfer learning and fine-tuning

In this paper, we utilize transfer learning, accompanied by fine-tuning, to improve model adaptability and convergence, enabling efficient knowledge transfer from a source dataset to a target dataset. Fine tuning is applied to models pre-trained on large datasets to effectively adapt and perform well even when tuned with comparatively smaller datasets. In this way, we leverage the learned features from the large dataset, applying them to a smaller, possibly more specific dataset, to enhance learning efficiency and performance.

In our methodology, the optimization process begins with training the source model, which is formalized as follows:

$$\Theta_s^* = \underset{\Theta_s}{\operatorname{argmin}}\, \mathcal{L}(M_s(\Theta_s), S). \tag{1}$$

Eq. (1) describes the process of iteratively updating the parameters Θ_s of the source model M_s to minimize the loss function $\mathcal{L}$ over the source dataset S. The best parameters, Θ_s^* are achieved at the end of this training phase and serve as the initial settings for the subsequent deep tuning phase applied to the target model. This sequential approach ensures that the source model's insights are not discarded but rather enhanced to suit the new data context represented by dataset T. Thus, the transition from the source model to the target model involves an initial parameter transfer followed by fine-tuning, as outlined in Eq. (2).

$$\Theta_t^* = \underset{\Theta_t}{\operatorname{argmin}}\, \mathcal{L}(M_t(\Theta_t), T). \tag{2}$$

Here, T represents the target dataset, and $\mathcal{L}$ is the loss function specifically adapted to the target's requirements. In Eq. (2), the fine-tuning starts from the parameter set Θ_s^*, thus leveraging the pre-trained state to accelerate and refine the learning process on T. This method is particularly effective for scenarios where the source and target datasets are related but distinct enough to require fine-tuning, such as in domain adaptation tasks.

Specifically, for binary classification scenarios, we employ a binary cross-entropy loss:

$$\left[\mathcal{L}(y, y') = -[y \cdot \log(y') + (1 - y) \cdot \log(1 - y')]\right], \tag{3}$$

where y is the ground truth label (0 for benign, 1 for DDoS) and y' is the predicted probability of DDoS by the model. In multi-class classification, we have employed a categorical cross-entropy loss:

$$\left[\mathcal{L}(y, y') = -\sum [y \cdot \log(y')]\right], \tag{4}$$

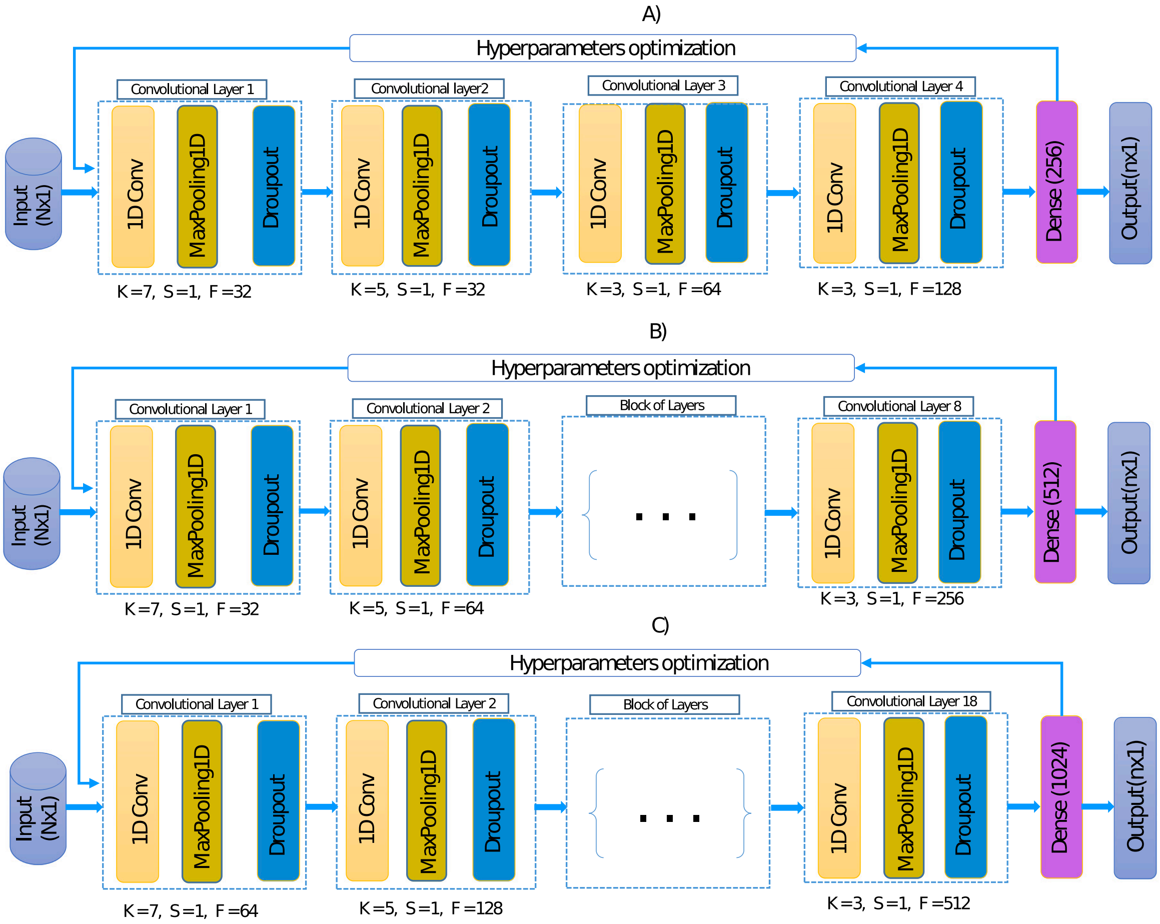

Fig. 2. Overview of custom 1D CNN models: (A) Conv4, (B) Conv8, and (C) Conv18. These models feature ReLU activation in their internal layers to introduce non-linearity and use either Softmax (for multi-class classification) or Sigmoid (for binary classification) in the output layer. Designed to manage varying complexities, the models range from 4 to 18 convolutional layers, optimizing them for efficient feature extraction from a 1D input dataset. Tailored for both simple and complex DDoS attack classifications, the models are defined by the kernel (K), stride (S), and feature map size (F) and have undergone hyperparameter optimization.

where $\mathcal{L}(y, y')$ is the categorical cross-entropy loss, y is a one-hot encoded vector representing the true class labels, and y' is a vector of predicted class probabilities produced by the model.

3.4. Hyperparameter optimization

Hyperparameters play a critical role in determining the model's performance and effectiveness. The following hyperparameters were selected and tuned for optimal results: learning rate, batch size, dropout rate, regularization parameters (L1 and L2), and number of layers. The rationale behind the selection of these hyper-parameters stems from their significant impact on the model's performance and generalization ability. By tuning these hyperparameters, we aim to achieve the best trade-off between accuracy, computational efficiency, and model robustness. Additionally, we consider the specific requirements of DDoS detection in cybersecurity, including the diverse range of attack scenarios and the distinct characteristics of network traffic, when determining the optimal hyperparameter values.

When it comes to hyperparameter optimization, several techniques can be employed, including random search, grid search, Bayesian optimization, and hyperband. Hyperband improves on random search by efficiently prioritizing configurations using explore–exploit principles, allocating resources more effectively to find the best settings. In this paper, we have used the hyperband keras library for hyperparameter tuning. We opted for this approach due to its well-balanced trade-off between time, resource utilization, and performance.

In this paper, we have employed a four-step approach for fine-tuning and hyperparameter optimization in our models. *(i) Model definition.* We select and define the specific DL architecture tailored to our dataset, establishing the foundation for our optimization process. *(ii) Hyperparameter selection.* We identify the hyperparameters for tuning, specific to the chosen DL architecture. *(iii) Search space definition.* We establish the search space for each hyperparameter by specifying their possible range or values, *(iv) Search algorithm specification.* We apply the hyperband search algorithm to efficiently navigate the hyperparameter space.

We executed the algorithm specified in Algorithm 1 by utilizing the defined search space. In this context, units refer to the number of neurons in a given layer of our neural network model.

4. Experimental results

4.1. Databases used and preprocessing

To evaluate the performance of our proposed adaptive transfer learning models, we selected four well-known datasets in cyber security: KDDCup'99 (Bay et al., 2000), UNSW-NB15 (Moustafa and Slay, 2015), CSE-CIC-IDS2018 (Sharafaldin et al., 2018), and CIC-DDoS2019 (Sharafaldin et al., 2019). These datasets are widely recognized as industry benchmarks in the domain of cybersecurity (Gümüşbaş et al., 2020; Sharafaldin et al., 2017). They encompass a wide spectrum of attack scenarios, providing us with the means

Algorithm 1: Adaptive hyperparameter optimization for DDoS attack detection

```
Input: Preprocessed dataset df
Output: Optimized model with the best hyperparameter combinations
Initialization: Initialize the model;
Define Search Space and Tuner: Hyperband tuner;
Hyperparameter Tuning: Perform hyperparameter tuning;
for $i$ in specified epoch range do
    for batch size ($bs$) from 16 to 512 do
        for dropout rate ($dr$) from 0.1 to 0.5 do
            Learning Rate Variation: If learning rate ($LR$) is between 0.001 and 0.1;
                Test different learning rates within the specified range;
            Unit Variation: For each number of units [32, 64, 128, 256, 512];
                Experiment with different unit configurations;
            Hyperband Search: Apply Hyperband search algorithm to identify best model configuration;
Best Hyperparameter Combination: Retrieve the best hyperparameter combination that resulted in the highest performance;
Train Model with Best Hyperparameters: Retrain the model using the identified best hyperparameter combination;
Save Optimized DL Model: Store the optimized model for future use;
```

to effectively train DL models to detect a variety of attack types. Specifically, we chose UNSW-NB15 for its realistic network traffic patterns, KDDCup'99 for its comprehensive set of network intrusions, CSE-CIC-IDS2018 for its modern attack and traffic types, and CIC-DDoS2019 for its detailed DDoS attack scenarios. This diversity allows us to evaluate the robustness and efficacy of models across different types of network environments and attack vectors. In the following, we delve into detailed explanations of these datasets and the corresponding preprocessing.

KDDCup'99 dataset (Bay et al., 2000). The KDDCup'99 dataset was specifically created for the KDDcup 1999 competition which aimed to develop effective methods for detecting unauthorized access and malicious activities in computer networks. This dataset includes an extensive collection of network connection records, approximately 5 million entries. This dataset comprehensively includes both normal connections and 22 types of cyber-attacks, classified into four major categories. These attacks consist of DoS-based (back, LAND, ping of death, teardrop, Neptune, and smurf attacks), U2R (buffer overflow, load module, perl, and rootkit attacks), R2L (ftp-write, guess-password, imap, multihop, PHF, spy, warezclient, and warezmaster attacks), and probe-based (port sweep, IP sweep, NMAP, and Satan attacks). Each network connection record is characterized by 42 features (Aggarwal and Sharma, 2015).

We performed data preprocessing for this dataset following the procedures outlined in Section 3. Initially, we converted the categorical data into numeric values. Next, we normalized the entire dataset using the min–max normalization method to scale the data within a standardized range of 0 to 1. To enhance data quality, we identified and removed duplicate rows, NaN values, missing values, and columns containing only zero values. After conducting normalization and data quality enhancement procedures, the dataset consists of 494,020 rows and 42 features.

UNSW-NB15 dataset (Moustafa and Slay, 2015). This dataset contains 9 unique attack types and 49 features. The attack categories consist of Analysis, Fuzzers, Backdoors, DoS, Exploits, Reconnaissance, Generic, Shellcode, and Worms. These attack types cover a wide range of cyber threats, enabling a thorough assessment of IDS. After preprocessing, we retained 642,566 rows and 45 features for further analysis.

CSE-CIC IDS2018 dataset (Sharafaldin et al., 2018). The dataset records network traffic in a controlled lab environment, capturing both benign traffic and seven distinct cyberattack scenarios. The attacking infrastructure involves 50 machines, while the victim organization consists of 5 departments, comprising 420 machines and 50 servers. The dataset consists of captured network traffic and system logs from each machine (Sharafaldin et al., 2018). This dataset encompasses diverse attack scenarios, including DoS, DDoS, port scanning, and malicious code activities. To support ML algorithms, the dataset creators have specifically processed a version tailored for this purpose. This processed version is accessible as a set of CSV files, incorporating 80 features extracted from the captured traffic using CICFlowMeter-V3. This paper focuses specifically on segments of the dataset related to DDoS and benign traffic. The dataset contains information about seven types of DDoS attacks: GoldenEye, Slowloris, Hulk, SlowHTTPTest, LOIC-HTTP, HOIC, LOIC-UDP, and benign network traffic.

We performed the preprocessing and discovered and removed duplicate rows in the dataset, eliminating 3,708,162 redundant entries. Additionally, we removed 17 columns, which comprised socket-related features and only zero values. After conducting normalization and data quality enhancement procedures, the dataset consists of 7,384,563 rows and 66 features. Fig. 3 presents samples of the converted images from each class, ranging from Class C0 to C7. Class C0 represents benign traffic, while classes C1 to C7 represent different types of attack traffic.

CIC-DDoS2019 dataset (Sharafaldin et al., 2019). The dataset offers comprehensive data on various DDoS attack vectors, including UDP flood, TCP SYN flood, and HTTP flood. These specifics facilitate a nuanced analysis of distinct attack characteristics. The CIC-DDoS2019 dataset encompasses 18 types of attacks, including both reflection- and exploitation-based attacks such as DrDoS-LDAP, DrDoS-MSSQL, DrDoS-NetBIOS, DrDoS-NMP, DrDoS-SSDP, DrDoS-UDP, UDP-lag, WebDDoS, Syn, TFTP, DrDoS-DNS, DrDoS-NTP, Portmap, NetBIOS, LDAP, MSSQL, UDP, and UDPLag. To streamline the dataset for multi-class classification, we merged similar attacks based on their attack techniques, network behaviors, and naming conventions. For instance, different types of UDP-based attacks — DrDoS-UDP, UDP, UDP-lag, and UDPLag — were grouped due to their shared characteristic of overwhelming the target with excessive requests. This merging process, which aligns with existing practices in the literature (Akgun et al., 2022) simplifies the dataset without compromising the integrity of the attack patterns, thereby enhancing the manageability and training efficiency of models.

Consequently, the dataset now profiles 12 distinct attack types: TFTP, UDP, NTP, SSDP, SYN, MSSQL, SNMP, DNS, BENIGN, LDAP, NetBIOS, Portmap, and WebDDoS. Table 1 details the dataset's cardinality, while Fig. 4 illustrates samples of the converted images from each class in these datasets.

During preprocessing, we removed 59,936,580 rows and 20 columns filled predominantly with zero values and socket-related features, which lacked variability, reducing the dataset to 10 million rows and 66 columns.

4.2. Model evaluation and selection

The following evaluation metrics were applied in this study.

- *Error (ERR).* The proportion of incorrect classifications to total observations

$$ERR = \frac{FP + FN}{TP + TN + FP + FN}. \tag{5}$$

- *Accuracy (ACC).* The percentage of exact predictions out of the total instances.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = 1 - ERR. \tag{6}$$

- *Precision (PR).* Also known as false negative rate (FNR), it is the ratio of correct positive predictions (TP) to the total positive predictions of the model.

$$PR = \frac{TP}{TP + FP}. \tag{7}$$

Table 1
Traffic types and their cardinality from the preprocessed CIC-DDoS2019, CSE-CIC-IDS2018, UNSW-NB15, and KDDcup'99 datasets.

CIC-DDoS2019		CSE-CIC-IDS2018		UNSW-NB15		KDDCup'99	
Traffic type	Cardinality	Traffic type	Cardinality	Traffic type	Cardinality	Traffic type	Cardinality
TFTP	4,396,770	Benign	6,412,040	Benign	321,283	Benign	97,280
UDP	2,510,574	DoS attacks-GoldenEye	41,406	Generic	215,481	DoS	391,457
NTP	1,112,902	DoS attacks Slowloris	9,908	Exploits	44,525	U2R	52
SSDP	891,220	DDoS attacks-LOIC-HTTP	575,364	DoS	16,353	R2L	1,124
SYN	687,524	DDoS attacks-LOIC-UDP	1,730	Reconnaissance	13,987	Probe	4,107
MSSQL	484,070	DDoS attacks-HOIC	198,861	Fuzzers	24,246	–	–
SNMP	114,179	DDOS attacks-slowHTTP Test	55	Analysis	2,677	–	–
DNS	113,252	DoS attacks-HULK	145,199	Backdoor	2,329	–	–
BENIGN	99,154	–	–	Shellcode	1,511	–	–
LDAP	48,469	–	–	Worms	174	–	–
NetBIOS	31,052	–	–	–	–	–	–
Portmap	1,638	–	–	–	–	–	–
WebDDoS	414	–	–	–	–	–	–
Total	*10,491,218*	*Total*	*7,384,563*	*Total*	*642,566*	*Total*	*494,020*

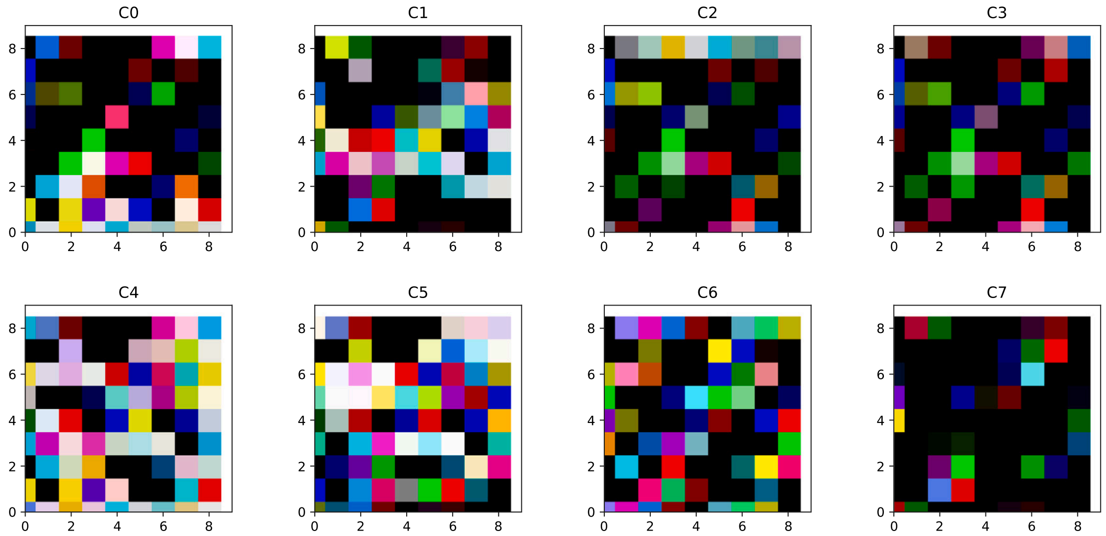

Fig. 3. Representative samples showcasing converted images from each category, ranging from Class 0 to Class 7, derived from the CSE-CIC-IDS2018 dataset. The classes include: Benign Traffic (C0), DDoS Attacks - LOIC-HTTP (C1), DDoS Attack - HOIC (C2), DoS Attacks - Hulk (C3), DoS Attacks - GoldenEye (C4), DoS Attacks - Slowloris (C5), DDoS Attack - LOIC-UDP (C6), and DoS Attacks - SlowHTTPTest (C7).

- *Recall (REC)*. Also known as detection rate (DR) or true positive rate (TPR), it is the percentage of correct positive predictions (TP) on the total of positive instances.

$$REC = \frac{TP}{TP + FN}. \tag{8}$$

- *F-Score (FS)*. Also known as f1-score, it is the harmonic mean of the precision and recall metrics. It is especially useful when class distribution is imbalanced:

$$FS = \frac{2 \cdot \mathrm{REC} \cdot \mathrm{PR}}{\mathrm{REC} + \mathrm{PR}}. \tag{9}$$

4.3. Results and discussion

In this section, we evaluate the performance of our adaptive transfer learning approach across different datasets DL models, including CNN architectures, along with fine-tuning pre-trained models for DDoS attack detection. We thoroughly examine the results of the capabilities of both DL and transfer learning models in DDoS attack detection. The experiments were performed using Google Colab Pro, with GPU enabled and RAM set to "high". For data preprocessing and experimentation, we used Python with libraries PIL, Dask, Pandas, Keras, and Sci-Kit Learn. We partitioned the dataset into three segments: 40% for training, 20% for validation, and the remaining 40% for testing.

As shown in Table 2, we define the search space for hyperparameter learning rates as [1e−3, 1e−4, 1e−5], to find the value that ensures efficient convergence without causing overshooting or slow convergence. For batch size, common values range from 16 to 512, and finding the optimal batch size can impact training speed and weight updates. We tune the dropout rate between 0.1 and 0.5 to prevent overfitting while preserving useful information. Moreover, we test different activation functions, ReLU, Sigmoid, or Tanh, to identify the one that allows the model to capture non-linear relationships. We adjust the number of hidden layers and neurons in each layer to find the optimal balance between model complexity and generalization ability. We considered layer configurations [32, 64, 128] or [64, 128, 256, 512] and evaluated their impact on performance. We also considered L1 and L2 regularization techniques to find the best trade-off between reducing over-fitting and model performance.

Table 2
Defined hyperparameters and search space ranges for hyperparameter tuning strategies.

Hyperparameters	Hyperparameter values/search space
Activation function	*ReLU, Tanh, Sigmoid*
Units	32, 64, 128, 512
Dropout rate	0.1 – 0.5
Learning rate	[1e−3, 1e−4, 1e-5]
Epoch	[10, 20, 30, 40]
Batch size	16 – 512
Regularization	L1, L2

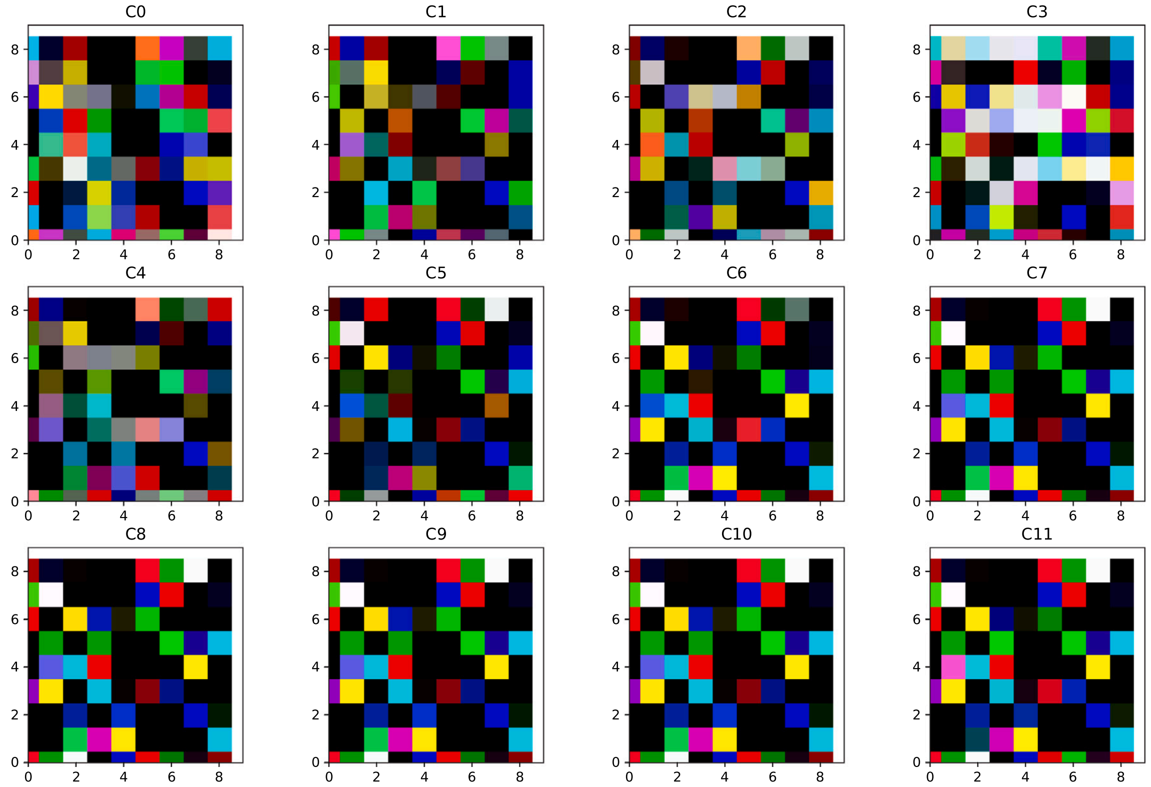

Fig. 4. Representative samples showcasing converted images from each category ranging from Class 0 to Class 11, including benign traffic from the CIC-DDoS2019 dataset. The classifications are as follows: Benign (C0), NTP (C1), TFTP (C2), Syn (C3), UDP (C4), MSSQL (C5), DNS (C6), LDAP (C7), DrDoS_SNMP (C8), NetBIOS (C9), SSDP (C10), and WebDDoS (C11).

Table 3
Accuracy results of transfer learning source model for binary and multiclass classification.

Variants	Accuracy			
	Dataset	Conv4	Conv8	Conv18
Binary	CIC-DDoS2019	99.90	**99.94**	99.88
	CSE-CIC-IDS2018	**99.82**	99.81	99.73
	UNSW-NB15	**98.78**	98.71	98.35
	KDDCup'99	99.42	**99.67**	99.59
Multi	CIC-DDoS2019	**91.99**	91.84	91.76
	CSE-CIC-IDS2018	**99.84**	99.82	96.98
	UNSW-NB15	97.51	97.55	**97.61**
	KDDCup'99	95.12	95.95	**95.95**

4.4. Customized CNN and pre-trained models transfer learning results

CNN customized model. We trained the custom CNN models using the Adam optimizer. The loss functions were categorical cross-entropy for multiclass classification and binary cross-entropy for binary classification.

Conv4 achieved an accuracy of 99.90%, Conv8 recorded 99.94%, and Conv18 reached 99.88% in identifying benign versus DDoS attack traffic within the CIC-DDoS2019 dataset. For multi-class classification of specific attack types, Conv4 and Conv8 demonstrated accuracies of 99.84% and 99.82%, respectively, on the CSE-CIC-IDS2018 dataset, while Conv18 achieved 97.61% on the UNSW-NB15 dataset (see Table 3).

To explore the transferability and adaptability of models trained on specific networks or datasets to new and diverse environments, we assess their performance by applying them to various target datasets. The target datasets used in this evaluation CSE-CIC-IDS2018, CIC-DDoS2019, KDDCup'99, and UNSW-NB15 enable a comprehensive assessment of the models' adaptability across diverse network environments.

Table 4
Accuracy evaluation of models in detecting benign to attack traffic detection tasks transferred to various target datasets.

Dataset	Dataset	Target dataset			
		CIC-DDoS2019	CSE-CIC-IDS2018	UNSW-NB15	KDDCup'99
CICDDoS 2019	Conv4		99.81	99.78	99.73
	Conv8		99.92	99.67	99.88
	Conv18		**99.99**	**99.92**	**99.94**
CSE-CIC-IDS2018	Conv4	99.67		99.32	99.48
	Conv8	99.81		99.46	99.74
	Conv18	**99.88**		**99.46**	**99.78**
UNSW-NB15	Conv4	98.82	99.21		98.24
	Conv8	**98.88**	99.23		**98.46**
	Conv18	98.73	**99.42**		98.33
KDD cup99	Conv4	96.25	**99.04**	95.24	
	Conv8	**97.02**	98.33	96.34	
	Conv18	96.66	98.13	**98.46**	

Initially trained on the CIC-DDoS2019 dataset, the source model demonstrated robust adaptability across various target datasets. In binary classification tasks, the Conv18 model, transferred from CIC-DDoS2019 to the CSE-CIC-IDS2018 dataset, achieved an impressive 99.99% accuracy in distinguishing benign from DDoS network traffic. Refer to Table 4 for detailed results.

The proposed model exhibits a consistent adaptation across source to target dataset transfers, demonstrating minimal differences in binary classification performance. This underscores the model's robust adaptability across various datasets. Additionally, the model achieves better results compared to single-domain training. These findings explicitly confirm that our approach permits the achievement of greater accuracy relative to single-domain training.

Table 5
Accuracy evaluation of models in multiclass attack detection tasks transferred to various target datasets.

Dataset	Model	Target dataset			
		CIC-DDoS2019	CSE-CIC-IDS2018	UNSW-NB15	KDDCup'99
CIC-DDoS 2019	Conv4		99.83	98.04	96.98
	Conv8		99.91	98.23	98.76
	Conv18		**99.92**	**99.42**	**98.88**
CSE-CIC-IDS2018	Conv4	89.92		99.34	99.84
	Conv8	89.52		99.42	99.91
	Conv18	**93.62**		**99.84**	**99.96**
UNSW-NB15	Conv4	83.42	92.34		94.24
	Conv8	86.78	93.64		**94.46**
	Conv18	**88.92**	**93.86**		94.33
KDD cup99	Conv4	85.25	95.04	**96.84**	
	Conv8	85.54	**95.33**	96.04	
	Conv18	**85.66**	95.13	96.81	

Table 6
Performance metrics on various datasets and VGG16, VGG19, and ResNet50 pre-trained models for binary classification.

Dataset	Model	Accuracy %	Recall	Precision	F1-score
CIC-DDoS 2019	VGG16	**99.99**	**99.98**	**99.99**	**99.98**
	VGG19	99.99	99.96	99.97	99.98
	ResNet50	99.94	99.96	99.98	99.97
CSE-CIC-IDS2018	VGG16	99.99	99.99	99.98	99.98
	VGG19	**100.00**	**100.00**	**100.00**	**100.00**
	ResNet50	99.98	99.98	99.78	99.82
UNSW-NB15	VGG16	98.36	97.68	**98.70**	**98.86**
	VGG19	**98.64**	**98.68**	98.68	98.67
	ResNet50	98.60	98.62	98.67	98.65
KDD Cup'99	VGG16	99.69	**99.98**	**99.99**	99.98
	VGG19	**99.90**	99.96	99.97	**99.98**
	ResNet50	99.56	99.96	99.98	99.97

Table 7
Performance metrics on various datasets and VGG16, VGG19, and ResNet50 pre-trained models for multi-class classification.

Dataset	Model	Accuracy %	Recall	Precision	F1-score
CIC-DDoS 2019	VGG16	92.19	92.56	92.30	92.28
	VGG19	**92.65**	**92.29**	**92.24**	**92.91**
	ResNet50	91.71	91.21	91.82	91.03
CSE-CIC-IDS2018	VGG16	99.21	99.21	99.21	99.21
	VGG19	**99.97**	**99.98**	**99.98**	**99.98**
	ResNet50	99.81	99.82	99.79	99.80
UNSW-NB15	VGG16	97.58	97.92	**98.68**	**98.42**
	VGG19	**97.59**	97.63	97.63	97.63
	ResNet50	97.36	97.23	97.04	97.64
KDD Cup'99	VGG16	97.46	97.23	97.49	97.98
	VGG19	**98.99**	**98.96**	98.97	**98.98**
	ResNet50	98.94	98.96	**98.98**	98.97

In multiclass classification, the transfer of the Conv18 model from CIC-DDoS2019 to CSE-CIC-IDS2018 yielded a performance of 99.92%, while the reverse transfer achieved 93.62%, as detailed in Table 5. Comparing the present results to prior findings reveals a consistently high accuracy level of the models when transferred from CIC-DDoS2019 to other datasets, in both binary and multiclass tasks. These results suggest the model's effective adaptation to the target dataset's characteristics, particularly as dataset features increase.

In transferring a model from a dataset with fewer features and instances to a larger and more complex target dataset, we observed decreased accuracy values in specific attack type identification. For instance, Conv4 achieved a score of 83.42% when transferred from UNSW-NB15 to the CIC-DDoS2019 dataset. This can be attributed to significant dissimilarities in dataset characteristics, such as size and complexity, leading to challenges in the model's adaptation to diverse patterns. Conversely, when transferring a model trained on a larger and more complex dataset to a smaller and less complex target dataset, we observed improved accuracy. For instance, Conv18, when transferred from CIC-DDoS2019 to the KDDCup'99 dataset, demonstrated enhanced performance metrics.

The model's effectiveness largely arises from its robust capability to analyze and utilize feature patterns from the extensive source dataset. This capability enables it to adapt to the structurally simpler target dataset efficiently. Such flexibility demonstrates the model's capability to transfer knowledge effectively, especially from a well-labeled, larger dataset to a smaller one. This feature is precious for reducing the necessity of extensive data labeling while maintaining high accuracy in predictions on the target dataset.

Pre-trained models. In this experiment, we employed a transfer learning approach to leverage the capabilities of pre-trained ImageNet CNN architectures, specifically VGG16, VGG19, and ResNet50. The approach involved the transformation of network traffic data into image representations, a process visually illustrated in Fig. 3.

For the CSE-CIC-IDS2018 dataset, a subset of 41,883 images were selected, which depicted characteristics of either benign or malicious traffic. We then extended our analysis to distinguish between multiple types of DDoS attacks in addition to benign traffic. This required a more comprehensive set of images to adequately represent each class, resulting in the use of 269,616 images.

In the case of the CIC-DDoS2019 dataset, we had 78,368 images covering 12 different attack classes for training, testing, and validation. Sample images from this dataset are displayed in Fig. 4. In addition, we used 12,154 images from the KDDCup'99 dataset and 5629 images from the UNSW-NB15 dataset. In our experiment, we tailored pre-trained models for binary and multiclass DDoS attack detection. Additionally, as part of our comprehensive model optimization, we applied major hyperparameter adjustments across all models, including the frozen layer ranges in our framework.

The results presented in Table 6 demonstrate the binary classification efficacy of the VGG16, VGG19, and ResNet50 models in differentiating between benign and DDoS attack traffic. Notably, the VGG19 model achieves a score of 100% in accuracy, recall, precision, and F1-score on the CSE-CIC-IDS2018 dataset. Within the CIC-DDoS2019 dataset, VGG16, VGG19, and ResNet50 all demonstrate high accuracy, with scores of 99.99%, 99.99%, and 99.94%, respectively. For the KDDCup'99 and UNSW-NB15 datasets, VGG19 outperforms the others, achieving accuracy rates of 99.90% and 98.64%. These findings highlight VGG19's superior binary classification capabilities, especially in precisely identifying benign versus DDoS network attack traffic in various network scenarios.

The performance metrics detailed in Table 7 present a comprehensive evaluation of adaptive pre-trained models, including VGG16, VGG19, and ResNet50, applied to multi-class classification tasks across diverse datasets. Notably, VGG19 outperforms other models in multi-class classification efficiency. On the CSE-CIC-IDS2018 dataset, VGG19 achieves an accuracy of 99.97%. In the CIC-DDoS2019 dataset, it leads with an accuracy of 92.65%. For the KDDCup'99 dataset, VGG19 excels with 98.99% accuracy, slightly ahead of ResNet50, which scores 98.94%. Similarly, on the UNSW-NB15 dataset, VGG19 maintains strong performance, achieving an accuracy of 97.59%. These outcomes underscore the adaptability and superior effectiveness of VGG19 in handling multi-class classification challenges.

The VGG19 model consistently outperforms others across a range of datasets, demonstrating its adaptability in capturing complex patterns effectively. We found that VGG19's relatively simpler and shallower architecture is particularly effective in capturing essential textural features from the image-formatted data. Its use of uniformly small filter

Table 8
Comparison of proposed approach metrics with state-of-the-art methods by dataset and class variants.

Dataset	Class variants	References	Model	Accuracy (%)
CIC-DDoS2019	Binary	Doriguzzi-Corin et al. (2020)	CNN	99.87
		Our model	VGG19	99.99
	Multi-class	Chartuni and Márquez (2021)	DNN	81.77
		Agostinello et al. (2023)	CNN	77.29
		Our model	Conv18	93.62
CSE-CIC-IDS2018	Binary	Okey et al. (2023)	VGG16	98.80
		Our model	VGG19	100.00
	Multi-class	Agostinello et al. (2023)	CNN	99.88
		Our model	Conv18	99.92
UNSW-NB15	Binary	Du et al. (2023)	CNN-LSTM	94.43
		Our model	Conv18	99.92
	Multi-class	Almogren (2020)	DBN	96.34
		Our model	Conv18	99.84
KDDCup'99	Binary	Du et al. (2023)	CNN-LSTM	97.40
		Our model	ResNet50	99.32
	Multi-class	Wu et al. (2019)	TL-ConvNet	93.86
		Our model	VGG19	98.99

sizes might allow it to efficiently identify crucial, surface-level discriminative features. Although ResNet50 shows good performance, especially in the KDDCup'99 and UNSW-NB15 datasets, they also require more extensive training data to achieve optimal performance.

In binary classification tasks, transferred pre-trained models VGG19 using the CSE-CIC-IDS2018 dataset have scored higher accuracy results than traditional DL models. However, in multi-class classification, transferred custom CNN models, such as Conv18, demonstrate a distinct advantage. Moreover, the impact of transfer learning on model performance is particularly notable in the domains of IDS and DDoS attack detection.

To evaluate the efficacy of our proposed model, we conducted a thorough comparison with state-of-the-art DL and transfer learning models, across similar datasets. Our proposed Conv18 model achieved 99.92% accuracy in network attack identification, compared to a VGG-16 IDS that reached a 98.8% accuracy on the CSE-CIC-IDS2018 dataset, as reported by Okey et al. (2023). Additionally, the pre-trained VGG19 model exhibited 100% accuracy in distinguishing benign from DDoS network traffic in the CSE-CIC-IDS2018 dataset. The models presented in Agostinello et al. (2023) and Chartuni and Márquez (2021) achieved accuracy rates of 77.29% and 81.77%, respectively, on the CIC-DDoS2019 dataset for attack type classification. In contrast, our model surpassed these results, achieving an accuracy of 93.62%. Additionally, in Wu et al. (2019), the TL-ConvNet model for the KDDCup'99 dataset demonstrated an accuracy of 93.86%, while our adaptive pretrained VGG19 model achieved a significantly higher accuracy of 98.99%. Furthermore, the Deep Belief Network (DBN) model by Almogren (2020) achieved a 96.34% accuracy for the UNSW-NB15 dataset, with our Conv18 model achieving 99.84% accuracy in detecting specific attack types. As detailed in Table 8, these findings indicate that our model performs well in comparison to existing approaches, particularly in DDoS attack detection and specific attack types identification, demonstrating the effectiveness of our employed adaptive transfer learning techniques.

5. Conclusions

DDoS attacks pose significant challenges to organizations worldwide, with their disruptive impact on network infrastructure availability and integrity. Building attack detection systems based on DL holds the promise of achieving high accuracy in detecting attack patterns in network traffic data. However, a major difficulty in developing DL-based IDS is the scarcity of large, labeled datasets that accurately represent today's network environments. In this paper, we proposed an adaptive transfer learning framework with fine-tuning and hyperparameter optimization. We employed custom CNN models (Conv4, Conv8, and Conv18), along with pretrained models (VGG16, VGG19, and ResNet50), trained on cybersecurity benchmark datasets, including KDDCup'99, UNSW-NB15, CSE-CIC-IDS2018, and CIC-DDoS2019.

Our experiments compared the performance of models trained with and without transfer learning in network traffic classification. The pretrained VGG19 model excelled in binary classification, effectively separating benign from malicious network traffic. Our custom-transferred Conv18 model achieved better accuracy, precision, recall, and F1-measure in detecting attack types, particularly in multi-label classification scenarios. Comparison of the current results with prior findings reveals a consistently high accuracy level of the models when transferred from the CIC-DDoS2019 dataset to others, in both binary and multiclass tasks. These results suggest the models' effective adaptation to the characteristics of the target dataset, especially as the number of dataset features increases. This shows that transfer learning proves to be a valuable approach to enhancing DDoS attack detection, even with limited labeled data.

Future work will enhance the practicality and robustness of DL and transfer learning models by prioritizing diverse dataset evaluation, defense against adversarial attacks, real-time implementation, and scalability.

Acknowledgments

This work was supported in part by the EC under projects Chips JU EdgeAI (101097300) and GLACIATION (101070141), and by project SERICS (PE00000014) under the MUR NRRP funded by the EU – NextGenerationEU. We also thank the NVIDIA Corporation for the GPU donated. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the Italian MUR. Neither the European Union nor Italian MUR can be held responsible for them.

References

Aggarwal, P., Sharma, S.K., 2015. Analysis of KDD dataset attributes-class wise for intrusion detection. Procedia Comput. Sci. 57, 842–851.

Agostinello, D., Genovese, A., Piuri, V., 2023. Anomaly-based intrusion detection system for DDoS attack with deep learning techniques. In: Proceedings of the 20th International Conference on Security and Cryptography. 1. SCITEPRESS, pp. 267–275.

Agrawal, N., Tapaswi, S., 2019. Defense mechanisms against DDoS attacks in a cloud computing environment: State-of-the-art and research challenges. IEEE Commun. Surv. Tutor. 21 (4), 3769–3795.

Akgun, D., Hizal, S., Cavusoglu, U., 2022. A new DDoS attacks intrusion detection model based on deep learning for cybersecurity. Comput. Secur. 118, 102748.

Almogren, A.S., 2020. Intrusion detection in edge-of-things computing. J. Parallel Distrib. Comput. 137, 259–265.

Bay, S.D., Kibler, D., Pazzani, M.J., Smyth, P., 2000. The UCI KDD archive of large data sets for data mining research and experimentation. ACM SIGKDD Explor. Newsl. 2 (2), 81–85, https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. (Accessed: (July 21/2023)).

Chadd, A., 2018. DDoS attacks: past, present and future. Netw. Secur. 2018 (7), 13–15.

Chartuni, A., Márquez, J., 2021. Multi-classifier of DDoS attacks in computer networks built on neural networks. Appl. Sci. 11 (22), 10609.

Chen, J., Yang, Y.-t., Hu, K.-k., Zheng, H.-b., Wang, Z., 2019. DAD-MCNN: DDoS attack detection via multi-channel CNN. In: Proceedings of the 2019 11th International Conference on Machine Learning and Computing. pp. 484–488.

Cheng, J., Zhang, C., Tang, X., Sheng, V.S., Dong, Z., Li, J., 2018. Adaptive DDoS attack detection method based on multiple-kernel learning. Secur. Commun. Netw. 2018, 1–19.

Cil, A.E., Yildiz, K., Buldu, A., 2021. Detection of DDoS attacks with feed forward based deep neural network model. Expert Syst. Appl. 169, 114520.

Das, A., et al., 2022. A deep transfer learning approach to enhance network intrusion detection capabilities for cyber security. Int. J. Adv. Comput. Sci. Appl. 13 (4).

Diro, A.A., Chilamkurti, N., 2018. Distributed attack detection scheme using deep learning approach for internet of things. Future Gener. Comput. Syst. 82, 761–768.

Doriguzzi-Corin, R., Millar, S., Scott-Hayward, S., Martinez-del Rincon, J., Siracusa, D., 2020. LUCID: A practical, lightweight deep learning solution for DDoS attack detection. IEEE Trans. Netw. Serv. Manag. 17 (2), 876–889.

Du, J., Yang, K., Hu, Y., Jiang, L., 2023. NIDS-CNNLSTM: Network intrusion detection classification model based on deep learning. IEEE Access 11, 24808–24821.

Elsaeidy, A.A., Jamalipour, A., Munasinghe, K.S., 2021. A hybrid deep learning approach for replay and DDoS attack detection in a smart city. IEEE Access 9, 154864–154875.

Gümüşbaş, D., Yıldırım, T., Genovese, A., Scotti, F., 2020. A comprehensive survey of databases and deep learning methods for cybersecurity and intrusion detection systems. IEEE Syst. J. 15 (2), 1717–1731.

Hnamte, V., Hussain, J., 2023. DCNNBiLSTM: An efficient hybrid deep learning-based intrusion detection system. Telematics Inform. Rep. 10, 100053.

Jia, Q., Wang, H., Fleck, D., Li, F., Stavrou, A., Powell, W., 2014. Catch me if you can: A cloud-enabled ddos defense. In: 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks. pp. 264–275.

Kansal, V., Dave, M., 2017. DDoS attack isolation using moving target defense. In: 2017 International Conference on Computing, Communication and Automation (ICCCA). pp. 511–514.

Kolias, C., Kambourakis, G., Stavrou, A., Voas, J., 2017. DDoS in the IoT: Mirai and other botnets. Computer 50 (7), 80–84.

Kushwah, G.S., Ranga, V., 2021. Optimized extreme learning machine for detecting DDoS attacks in cloud computing. Comput. Secur. 105, 102260.

Masum, M., Shahriar, H., 2021. A transfer learning with deep neural network approach for network intrusion detection.. Int. J. Intell. Comput. Res. 12.1.

Moustafa, N., Slay, J., 2015. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS). IEEE, pp. 1–6.

Nugraha, B., Murthy, R.N., 2020. Deep learning-based slow DDoS attack detection in SDN-based networks. In: 2020 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN). IEEE, pp. 51–56.

Ogbuanya, C.E., 2021. Improved dimensionality reduction of various datasets using novel multiplicative factoring principal component analysis (MPCA). Int. J. Comput. Commun. Eng. 10 (4), 85–95.

Okey, O.D., Melgarejo, D.C., Saadi, M., Rosa, R.L., Kleinschmidt, J.H., Rodríguez, D.Z., 2023. Transfer learning approach to IDS on cloud IoT devices using optimized CNN. IEEE Access 11, 1023–1038.

Rodríguez, E., Valls, P., Otero, B., Costa, J.J., Verdú, J., Pajuelo, M.A., Canal, R., 2022. Transfer-learning-based intrusion detection framework in IoT networks. Sensors 22 (15), 5621.

Sabeel, U., Heydari, S.S., Mohanka, H., Bendhaou, Y., Elgazzar, K., El-Khatib, K., 2019. Evaluation of deep learning in detecting unknown network attacks. In: 2019 International Conference on Smart Applications, Communications and Networking (SmartNets). IEEE, pp. 1–6.

Shaaban, A.R., Abd-Elwanis, E., Hussein, M., 2019. DDoS attack detection and classification via convolutional neural network (CNN). In: 2019 Ninth International Conference on Intelligent Computing and Information Systems. ICICIS, IEEE, pp. 233–238.

Sharafaldin, I., Gharib, A., Lashkari, A.H., Ghorbani, A.A., 2017. Towards a reliable intrusion detection benchmark dataset. Softw. Netw. 2017 (1), 177–200.

Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A., 2018. Toward generating a new intrusion detection dataset and intrusion traffic characterization. ICISSp 1, 108–116.

Sharafaldin, I., Lashkari, A.H., Hakak, S., Ghorbani, A.A., 2019. Developing realistic distributed denial of service (ddos) attack dataset and taxonomy. In: 2019 International Carnahan Conference on Security Technology. ICCST, IEEE, pp. 1–8.

Venkatesan, S., Albanese, M., Amin, K., Jajodia, S., Wright, M., 2016. A moving target defense approach to mitigate ddos attacks against proxy-based architectures. In: 2016 IEEE Conference on Communications and Network Security. CNS, IEEE, pp. 198–206.

Vu, L., Nguyen, Q.U., Nguyen, D.N., Hoang, D.T., Dutkiewicz, E., 2020. Deep transfer learning for IoT attack detection. IEEE Access 8, 107335–107344.

Wei, Y., Jang-Jaccard, J., Sabrina, F., Singh, A., Xu, W., Camtepe, S., 2021. Ae-mlp: A hybrid deep learning approach for ddos detection and classification. IEEE Access 9, 146810–146821.

Wu, P., Guo, H., Buckland, R., 2019. A transfer learning approach for network intrusion detection. In: 2019 IEEE 4th International Conference on Big Data Analytics. ICBDA, IEEE, pp. 281–285.

Xue, B., Zhao, H., Yao, W., 2022. Deep transfer learning for IoT intrusion detection. In: 2022 3rd International Conference on Computing, Networks and Internet of Things. CNIOT, IEEE, pp. 88–94.

Yang, L., Shami, A., 2022. A transfer learning and optimized CNN based intrusion detection system for internet of vehicles. In: ICC 2022-IEEE International Conference on Communications. IEEE, pp. 2774–2779.

Yeom, S., Choi, C., Kim, K., 2022. LSTM-based collaborative source-side DDoS attack detection. IEEE Access 10, 44033–44045.

Zhang, Y., Liu, Y., Zhang, Y., Han, L., Zhao, J., Wu, Y., 2021. A DDoS attack detection method based on LSTM neural network in the internet of vehicles. In: Proceedings of the 4th International Conference on Information Technologies and Electrical Engineering. pp. 1–5.

Mulualem Bitew Anley is a Ph.D. student in the national Ph.D. program in Cybersecurity, jointly enrolled at the IMT School for Advanced Studies Lucca and the Università degli Studi di Milano, Italy. He received his MSc in Information Technology from University of Gondar, Ethiopia. His research interest focuses on the intersection of cybersecurity and artificial intelligence, with a particular emphasis on IoT and cloud security.

Angelo Genovese is an associate professor at the Computer Science Department, Università degli Studi di Milano, Italy. His main research interests are in signal and image processing, artificial intelligence applications, and design methodologies and algorithms for self adapting systems. He has been a visiting researcher with the University of Toronto, ON, Canada. He has published more than 70 papers in journals, conferences proceedings, and book. He Senior Member of IEEE. https://genovese.di.unimi.it

Davide Agostinello is deputy manager of Information Systems ICT Complex Structure at Papa Giovanni XXIII Hospital, Bergamo, Italy.

Vincenzo Piuri is a professor at the Computer Science Department, Università degli Studi di Milano, Italy. His main research interests are in artificial intelligence, intelligent systems, machine learning, pattern analysis and recognition, and dependability. He has been a visiting professor at The University of Texas at Austin, USA, and a visiting researcher at George Mason University, VA (USA. He has published more than 400 papers in journals, conference proceedings, and books. He is a Fellow of the IEEE, a Distinguished Scientist of ACM, and a Senior Member of INNS. https://piuri.di.unimi.it