Journal of Network and Computer Applications 221 (2024) 103784

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

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

Research paper

DTL-IDS: An optimized Intrusion Detection Framework using Deep Transfer Learning and Genetic Algorithm

Shahid Latif [a], Wadii Boulila [a,b,*], Anis Koubaa [a], Zhuo Zou [c], Jawad Ahmad [d,**]

[a] Robotics and Internet-of-Things Laboratory, Prince Sultan University, Riyadh 11586, Saudi Arabia
[b] RIADI Laboratory, National School of Computer Sciences, University of Manouba, Manouba, Tunisia
[c] School of Information Science and Engineering, Fudan University, Shanghai, China
[d] School of Computing, Engineering and the Built Environment, Edinburgh Napier University, Edinburgh EH10 5DT, United Kingdom

ARTICLE INFO

Dataset link: https://github.com/shahidlatif126/DTL-IDS-An-Optimized-Intrusion-Detection-Framework-using-DTL-and-GA

Keywords:
Cybersecurity
Genetic Algorithm
IIoT
Intrusion Detection
Transfer learning

ABSTRACT

In the dynamic field of the Industrial Internet of Things (IIoT), the networks are increasingly vulnerable to a diverse range of cyberattacks. This vulnerability necessitates the development of advanced intrusion detection systems (IDSs). Addressing this need, our research contributes to the existing cybersecurity literature by introducing an optimized Intrusion Detection System based on Deep Transfer Learning (DTL), specifically tailored for heterogeneous IIoT networks. Our framework employs a tri-layer architectural approach that synergistically integrates Convolutional Neural Networks (CNNs), Genetic Algorithms (GA), and bootstrap aggregation ensemble techniques. The methodology is executed in three critical stages: First, we convert a state-of-the-art cybersecurity dataset, Edge_IIoTset, into image data, thereby facilitating CNN-based analytics. Second, GA is utilized to fine-tune the hyperparameters of each base learning model, enhancing the model's adaptability and performance. Finally, the outputs of the top-performing models are amalgamated using ensemble techniques, bolstering the robustness of the IDS. Through rigorous evaluation protocols, our framework demonstrated exceptional performance, reliably achieving a 100% attack detection accuracy rate. This result establishes our framework as highly effective against 14 distinct types of cyberattacks. The findings bear significant implications for the ongoing development of secure, efficient, and adaptive IDS solutions in the complex landscape of IIoT networks.

1. Introduction

Industry 4.0 revolutionized smart manufacturing operations by incorporating many modern technologies, including the Internet of Things (IoT), artificial intelligence (AI), cloud and edge computing, big data analytics, robotics, and cybersecurity. Industry 4.0, also referred to as the IIoT, enhances the automation level in smart industries and warehouses (Sisinni et al., 2018). The IIoT is a pervasive network of interconnected devices that provide various computing services for smart industries. IIoT nodes can identify, process, and transfer useful information across different IIoT platforms. This allows for more efficient services and a better user experience in industries ranging from manufacturing to service supply. Along with several benefits, IIoT devices and networks are vulnerable to multiple types of cyberattacks because of their heterogeneous nature (Driss et al., 2021). It can cause multiple security and privacy issues in smart industries, which may lead to a huge financial loss in the worst-case scenario. Therefore, the enormous potential of IIoT cannot be realized without incorporating fast and robust cybersecurity mechanisms. Different types of cyberattacks can harm smart industries in multiple ways, such as unauthorized access to valuable consumer and industrial data, unavailability of IoT services, and damage the valuable equipment. An intrusion detection system (IDS) is considered a second-line defensive framework frequently used in conjunction with other security mechanisms to protect IIoT networks from cyberattacks (Latif et al., 2021; Moustafa et al., 2019; Khan et al., 2022).

Modern IDSs leverage traditional machine and deep learning (ML/DL) techniques to correlate the features from IoT traffic, identify the patterns in collected data, and detect the malicious traffic corresponding to attacks (Lee et al., 2021). Cybersecurity researchers spend a lot of time understanding these attacks and classifying them into well-known categories. However, the highly expandable nature of IIoT systems invites cybercriminals to try and find new ways to compromise an IIoT system. Therefore, a pre-defined list of attack classifications will not be able to respond to new and unique techniques

* Corresponding author at: Robotics and Internet-of-Things Laboratory, Prince Sultan University, Riyadh 11586, Saudi Arabia.
** Corresponding author.
E-mail addresses: wboulila@psu.edu.sa (W. Boulila), J.Ahmad@napier.ac.uk (J. Ahmad).

https://doi.org/10.1016/j.jnca.2023.103784
Received 16 February 2023; Received in revised form 4 September 2023; Accepted 2 November 2023
Available online 13 November 2023

used by intruders. The second major challenge is that most cybersecurity datasets are unbalanced, in which normal class samples contain a very high number compared to malicious samples. The uneven distribution of classes prevents the existing ML/DL-based IDSs from further improvements (Alzahem et al., 2022). To address these issues, a smart and robust system is required to deal with uneven distribution, auto-label the network alerts, and categorize them so the IDS analyst can focus on the specific alert type.

Deep transfer learning (DTL) can be a promising cybersecurity solution to overcome these issues, especially the problems related to training heterogeneous data of the IIoT. DTL utilizes the knowledge from pre-trained models to improve the performance of traditional DL models. DTL can alleviate the lack of labeled data for the targeted network by transferring information from a comparable circumstance with high-quality data (Yang and Shami, 2022; Li et al., 2021). Furthermore, the DTL can exchange knowledge even if the data characteristics of the source and target networks are not substantially similar. This article proposes an efficient and robust intrusion detection scheme using DTL. The proposed scheme is based on seven pre-trained convolutional neural network (CNN) architectures, including Xception, VGG16, VGG19, Inception, InceptionResNetV2, EfficientNetB7, and EfficientNetV2L. The optimum outcome of a CNN architecture also depends on selecting suitable hyperparameters. The proposed design's utilized pertained architectures are optimized and trained using a genetic algorithm (GA). After training, the five best-performing models are selected and integrated using the bootstrap aggregation ensemble technique. Finally, the efficiency of the proposed scheme is investigated using the latest IIoT security dataset Edge-IIoTset. The main contributions of the article are summarized in the following.

- First, the Edge-IIoTset dataset is preprocessed by removing unnecessary feature columns and encoding categorical data. After that, the complete preprocessed dataset is transformed into an image dataset, enabling easy learning to distinguish various cyberattack patterns.
- A highly efficient DTL-based intrusion detection scheme is proposed to detect a wide range of cyberattacks in IIoT networks. The proposed scheme is based on seven advanced pre-trained CNN architectures. Furthermore, all these architectures are optimally trained using GA. Finally, the outputs of the five best-performing models are integrated through the bootstrap aggregation ensemble technique.
- The proposed framework is rigorously evaluated through a comprehensive experimental performance analysis. The evaluation includes assessing the effectiveness of the framework in both binary and multiclass scenarios. This analysis provides valuable insights into the system's performance, demonstrating its robustness and efficacy in detecting a wide range of cyberattacks in IIoT networks.

The remainder of the article is structured as follows. Section 2 briefly examines the most recent cutting-edge transfer learning-based IDSs for IoT/IIoT networks. Section 3 presents the proposed framework, including the description of the dataset, its preprocessing, image transformation, training and testing of DL models, hyperparameter optimization, and ensembling. Section 4 contains the experimentation process, a brief discussion of results, and a comparison with existing studies. Finally, Section 5 concludes the research and presents some future directions.

2. Related work

Transfer learning (TL) is an emerging paradigm to develop and investigate the latest intrusion detection schemes for IoT/IIoT applications. This section briefly discusses some of the latest state-of-the-art TL-based schemes for cybersecurity applications.

Traditional IDSs are facing new and emerging challenges in terms of accuracy, efficiency, and robustness. Li et al. (2021) developed a TL-based scheme for intrusion detection in the Internet of Vehicles (IoV). The authors introduced a local update scheme that obtained pseudo-labels of unlabeled data in new attacks through pre-classifiers and utilized the pseudo-labeled data for multiple rounds of TL. The experimental results indicated that the proposed TL-based approach attained an accuracy of around 92%. Mehedi et al. (2022) developed a DTL-based IDS for IoT networks. The suggested scheme is based on an efficient attribute selection and dependable DTL-based ResNet model evaluation with real-world data. The efficiency of the designed model is investigated by conducting extensive experiments on the ToN_IoT dataset. Gou et al. (2009) presented a distributed transfer learning technique for intrusion detection in IoT. The suggested frameworks introduced TL into distributing network boosting algorithms for instructing the attack learning with poor performance. The proposed scheme is evaluated using KDD Cup 99 dataset and attained an attack detection accuracy of 97.3%. Mehedi et al. (2021) developed a DTL based IDS for In-Vehicle Network (IVN). The main contribution of this work is developing an attribute selection method to identify anomalous messages and accurately detect normal and malicious activities through the DTL-based LeNet model. The proposed scheme is investigated on a personal synthetic dataset and achieved an accuracy of 98.10%.

The advancement of DTL and its impactful outcomes in multiple fields has opened several paths to develop effective cybersecurity solutions for IoT networks. Bierbrauer et al. (2023) proposed a TL-based framework for network intrusion detection using raw network traffic. The authors utilized two datasets, UNSW-NB15, and CICIDS2017, to train one-dimensional CNN combined with a retained random forest model. The experimental results indicated that the proposed scheme attained an attack detection accuracy of 96.89%. Xu et al. (2021) designed a privacy-preserving multisource TL-based IDS. First, a Paillier homomorphic encryption scheme is utilized to encrypt the model trained from different resources. After that, a multisource TL-based IDS is proposed based on XGBoost. The experimental results demonstrated that the suggested approach effectively transferred encryption models from various source domains to the target domain with an accuracy of 93.01%. Singh et al. (2021) presented a DTL model to detect the darknet network from the network traffic data. The authors transformed the time-based features into color images to attain more accurate results and fed them into a pre-trained model for optimal feature extraction. The authors used ten pre-trained models with three baseline classifiers to identify the optimized pre-trained network. Experiment results illustrated that the VGG19 attained the highest attack detection accuracy of 96%. Abosata et al. (2023) proposed a federated-transfer-learning-assisted customized distributed IDS (FT-CID) model for intrusion detection in IoT. The proposed model contains three main stages: dataset collection, FTL-assisted edge IDS learning, and intrusion detection. In the dataset collection stage, the central server initializes the FT-CID model with a predefined model. It constructs a local model by observing the unique features of different low-power and lossy networks (RPL)-IoTs. Then, the edge IDSs are trained using local and globally shared parameters, updating them through transfer learning. As a result, the FT-CID achieved high RPL security by utilizing local and global parameters and attained an attack detection accuracy of 85.52%. Yan et al. (2023) presented TL-CNN-IDS, an intrusion detection system based on transfer and ensemble learning. The proposed approach involves preprocessing the dataset using IG-FCBF feature engineering methods and converting it into an image format suitable for input to CNN models. Three CNN models were selected as the base models, and the Tree-Structured Parzen Estimator algorithm was used for hyperparameter optimization. The optimized CNN model was then integrated using confidence averaging in ensemble learning. The proposed scheme attained impressive accuracy of 99.85% and 99.53% for the CICIDS2017 and NSL-KDD datasets, respectively.

Several research works have been proposed in the literature related to proposing novel DTL schemes for intrusion detection in IoT networks. However, the existing schemes discussed in the literature have a few limitations. First, most proposed schemes only focused on detecting a limited number of cyberattacks. Researchers utilized the old-generation datasets for the evaluation of their schemes. These datasets do not demonstrate the true IIoT environment. To overcome this issue, we selected the latest IIoT cybersecurity dataset Edge-IIoTset for our experiments. With this dataset, the proposed scheme successfully classified 14 classes of cyberattacks. The second major shortcoming of the existing research is the limited device profiling. In the proposed scheme, the utilized cybersecurity dataset considers the behavior of 10 IoT devices to enhance the integrity characteristics of the proposed model. Third, most studies consider one or a few pre-trained models in their schemes. This affected the attack detection accuracy and robustness of their schemes. The proposed scheme utilized 7 pre-trained CNN architectures and the hyperparameter optimization of each model through GA. After training, the five best-performing models were selected and integrated through the bootstrap aggregation ensemble technique. This technique increases the robustness and attack detection accuracy of the proposed framework.

3. The proposed framework

This section presents the overall research methodology of the proposed framework. First, it describes the dataset, data preprocessing, and transformation. After that, it briefly discusses the DTL, hyperparameter optimization, and ensembling. A block diagram of the proposed framework is presented in Fig. 1.

3.1. Dataset description

In the proposed framework, we utilized the latest, most realistic, and comprehensive cybersecurity dataset for IoT and IIoT applications. The Edge-IIoTset dataset is generated by Ferrag et al. (2022b) and can be publicly accessed from Ferrag et al. (2022a) for research purposes. This dataset was generated by developing a realistic IIoT environment containing more than 10 types of IoT devices, including temperature and humidity sensors, ultrasonic sensors, water level detectors, soil moisture sensors, flame sensors, pH sensors, and heart rate sensors. Moreover, it contains 14 attack categories related to IIoT connectivity protocols. The class distribution of the Edge-IIoTset is presented in Fig. 2.

3.2. Dataset preprocessing

Data preprocessing plays an important role in the optimal training of any ML/DL model. The original Edge_IIoTset dataset contains 62 features and 15 classes. After loading the dataset, first, we checked the NaN values in the dataset. This dataset has no NaN values. In the second stage, 815 duplicate rows are removed, which is just 0.037% of the total dataset. In the third stage, we removed some unnecessary feature columns that do not play any significant role in output predictions. These feature columns include "frame.time", "ip.dst_host", "ip.src_host", "arp.src.proto_ipv4", "http.file_data", "arp.dst.proto_ipv4", "http.request.full_uri", "http.request.uri.query", "icmp.transmit_timestamp", "tcp.payload", "tcp.options", "tcp.srcport", "udp.port", "tcp.dstport", and "mqtt.msg". After dropping these columns, the new dataset contains 1909671 rows and 47 columns. In the new dataset, the 8 features column contains object data type. We applied dummy encoding to 7 feature columns except "Attack_type" These columns include "http.request.method", "http.referer", "http.request.version", "dns.qry.name.len", "mqtt.conack.flags", "mqtt.protoname", and "mqtt.topic". After applying dummy encoding, the new dataset contains 1909671 rows and 96 columns. This dataset is finally exported as a CSV file.

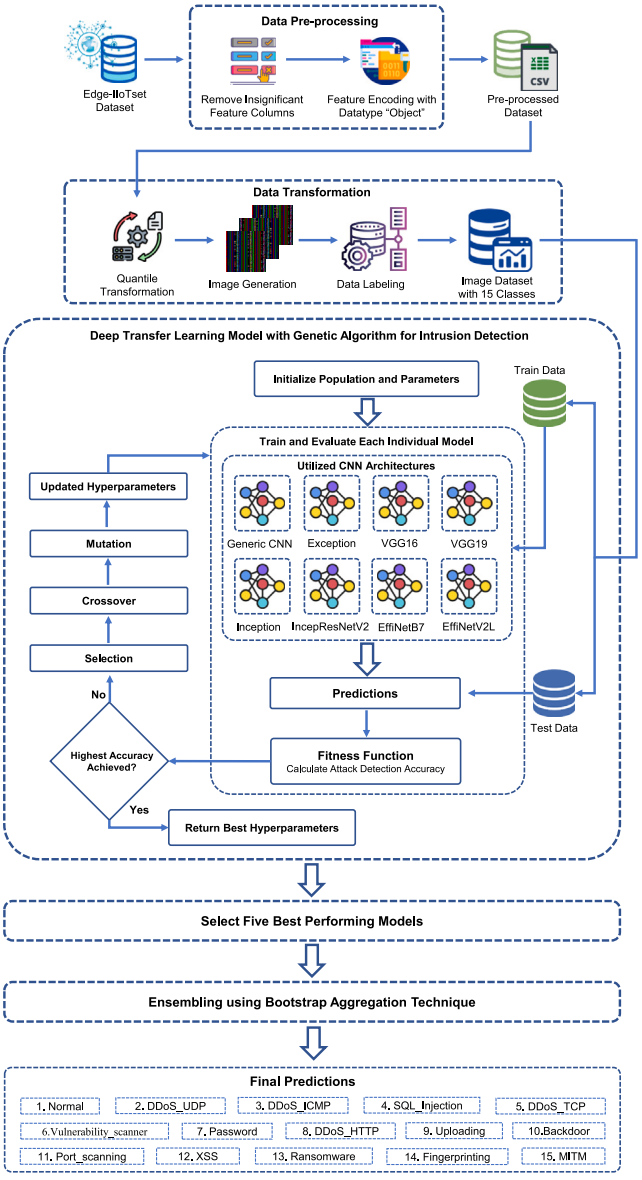


Fig. 1. Workflow of the proposed DTL intrusion detection framework.

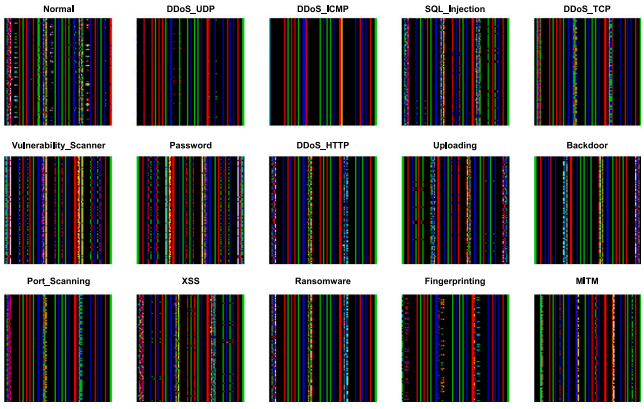


Fig. 2. Attacks distribution in Edge-IIoTset dataset.

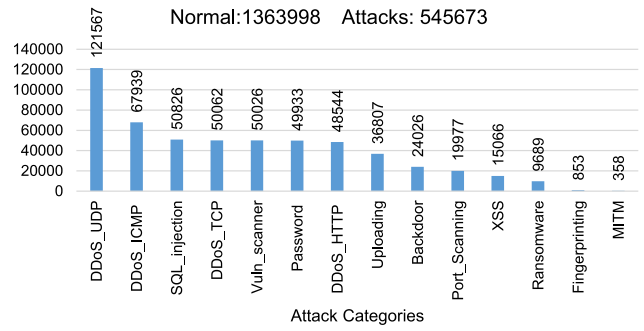


Fig. 3. Representative samples from the transformed Edge-IIoTset dataset.

3.3. Dataset transformation

In the proposed DTL model, we used the pre-trained CNN architectures. As CNN models generally work better for image-based datasets. The preprocessed Edge-IIoTset contains tabular data; we need to transform it into an image-based dataset. The first stage of data transformation is normalization. As image pixel values vary from 0 to 255, we need to map the preprocessed dataset in the range of 0 to 255. There are several techniques available for data normalization. In the designed IDS, the quantile normalization technique is utilized to transform the feature distribution to a normal distribution and recalculates all the feature values based on a normal distribution. In the results, most feature values remain close to the median values that efficiently handle the outliers (Lokman et al., 2020).

After normalization, the samples are transformed into blocks based on the feature sizes of the dataset. The preprocessed Edge-IIoTset dataset contains 95 columns. Each block of 285 successive samples with 95 columns is transformed into a color image shape of 95 × 95x3. After transformation, all the images are categorized based on the attack patterns. If all the samples in an image are normal, this image is labeled "Normal". If the image contains attack samples, it is labeled as this chunk's most frequent attack type. For Edge-IIoTset, images are labeled in14 attack categories including DDoS_UDP, DDoS_ICMP, SQL_injection, DDoS_TCP, Vulnerability_scanner, Password, DDoS_HTTP, Uploading, Backdoor, Port_Scanning, XSS, Ransomware, Fingerprinting, and MITM. The representative samples for each attack category in Edge-IIoTset are presented in Fig. 3.

3.4. Deep transfer learning with CNN

CNN is a widely used DL architecture for image recognition and classification applications. The images can be directly provided to the CNN architectures without further feature extraction and data reconstruction operations (Song et al., 2020). A generic CNN contains three layers: convolutional, pooling, and fully connected. The convolutional layer enables the automatic feature extraction from images through the convolution process. The pooling layers facilitate complexity reduction without compromising the important information using local correlations. Finally, a fully connected layer integrates all features and generates output.

DTL is the weight transfer of the DL model trained on one dataset to another (Shao et al., 2018). The DTL approach has been successfully employed for several image processing and classification tasks in the existing literature. The bottom layers of CNN models often learn basic feature patterns that apply to various tasks. The features acquired by the top layer of the model are specific to the dataset. Therefore, the bottom layers of CNN architectures can be directly utilized for various tasks (Shao et al., 2018). In addition, fine-tuning can increase DTL model performance. The majority of the layers of the pre-trained architectures are frozen throughout this process. Still, some of the top layers of these architectures are unfrozen to retain the model with new datasets. Fine-tuning allows the models to adjust the pre-trained architecture's higher-order features to match the targeted dataset better (Shao et al., 2018).

In the proposed DTL framework, seven CNN architectures, including Xception, VGG16, VGG19, Inception, InceptionResNetV2, EfficientNetB7, and EfficientNetV2L, are selected. These CNN architectures have been pre-trained on the ImageNet dataset and demonstrate promising results for generic image classification tasks. The ImageNet is a popular dataset for image processing applications with more than one million images of 1000 classes (Morid et al., 2021). VGG16 and VGG19 were developed and trained on ImageNet datasets by the University of Oxford researchers (Simonyan and Zisserman, 2014). The main difference between these two VGG models is the number of layers and parameters, including weights and biases. VGG16 is a 16-layer CNN, and VGG is a 19-layer CNN architecture. VGG19 has more layers and parameters, making it a more complex and powerful architecture and computationally intensive. Both of these models are widely utilized for multiple image classification tasks.

CNN incorporates inception modules to provide more efficient computing and deeper networks by dimensionality reduction with attacked 1 × 1 convolutions. Inception modules were specifically developed to solve the issues of high computational cost and overfitting. In CNN, these modules use various kernel filter sizes and arrange them to operate at the same level rather than stacking them sequentially (Szegedy et al., 2016). Xception is a deep CNN architecture with Depthwise Separable Convolutions (DSC). Google researchers introduced this architecture, which interprets inception modules in CNN as an intermediary step between ordinary convolution and DSC operation (Chollet, 2017). The resource requirements of the Xception model are somehow lower than the Inception model. Inception ResnetV2 is another Inception modification that combines Resnet's residual connections into the Inception network. This 164-layer architecture can categorize images into 1000 categories by learning rich feature representations from a large image dataset. EfficientNet is based on the baseline network created by the neural architecture search utilizing the AutoML MNAS framework (Tan and Le, 2019). It is a CNN architecture and scaling approach using a compound coefficient to consistently scale all depth/width/resolution dimensions. In contrast to current practice, which randomly scales these elements, the EfficientNet scaling technique uses a predetermined set of scaling factors to equally scale network width, depth, and resolution. EfficientNetV2 improves the EfficientNet by increasing training speed and parameter efficiency. This network was created by combining scaling with neural architecture search (Tan and Le, 2021). The primary objective is to maximize training speed and parameter efficiency.

3.5. Hyperparameter optimization

To ensure the optimum performance of the base models for the utilized dataset, the hyperparameters of CNN architectures must be optimized. The optimal selection of these hyperparameters can significantly impact the model's performance and efficiency. The traditional approach to finding the best hyperparameters for a specific dataset is the manual search, which depends on their experience training the DL models in solving similar problems. The major problem with manual search is that it can find the best parameters for a particular dataset and may not be the best settings for some other dataset. Therefore, manual selection cannot be an appropriate method for hyperparameter tuning in different experimental scenarios. More automated and guided techniques are required to investigate different configurations of the DL models for multiple problems.

The commonly used hyperparameter optimization (HPO) algorithms for CNNs in existing literature are Grid Search (GS), Random Search (RS), Bayesian Optimization (BO), and gradient-based optimization (GO) (Yu and Zhu, 2020). Despite having several advantages, these

techniques have a few drawbacks. GS usually requires dimensionality reduction to avoid excessive resource utilization (Yu and Zhu, 2020). The dimensionality reduction techniques usually compromise the effectiveness of the model. Compared to GS, the RS provides flexible resource allocation and easy parallelization. However, the RS is still a computationally expensive technique used for HPO to narrow the search space (Kim and Cho, 2019). BO has higher computational efficiency compared to GS and RS. Parallel computing is difficult to accomplish in BO since it is a sequential process in which a trial is proposed based on prior trial experience (Victoria and Maragatham, 2021). As a result, the hardware's performance cannot be completely leveraged to accelerate the solution process. GO is usually used for dealing with continuous hyperparameters and is not necessarily appropriate for dealing with discrete parameters. Furthermore, GO will increase the amount of forward and backward propagation calculations during the training phase, increasing the model's complexity (Zingg et al., 2008). Genetic Algorithm (GA) is a population-based optimization algorithm that speeds up the solution by simultaneously computing several populations (Natesha and Guddeti, 2021). In comparison to the aforementioned techniques, GA has no runtime constraints for data dimension, and as an optimization technique, it can minimize the number of model-solving iterations. The GA offers high parallelism, and model optimization parameters can be configured as discrete or continuous (Wu et al., 2022).

In this article, we utilized GA to obtain the best hyperparameters for the optimal training of CNN models. In GA-based HPO, each hyperparameter is represented by a chromosome. The value of the relevant hyperparameter is assigned to the decimal value of the representative chromosome. Several genes are encoded in binary format on each chromosome. The genes on this chromosome are then processed through selection, crossover, and mutation processes to determine the best parameters. Chromosomes with high fitness function values are more likely to be picked and passed on to the upcoming generation. These chromosomes produce new ones with the best characteristics of their parents in the next generation. Crossover is used to build new chromosomes by swapping the percentages of genes from various chromosomes representing the solution to acquire the mixing of solutions in the search space (Srinivas and Patnaik, 1994). Another operation used to make new chromosomes is a mutation, which involves randomly modifying one or more genes on a chromosome. Crossover and mutation processes ensure the diversity of subsequent generations, allowing them to have diverse features and reducing the possibility of missing valuable ones. The complete process of GA-based HPO is elaborated in algorithm 1.

Algorithm 1: Genetic Algorithm for Hyperparameter Optimization

Input population size: n
maximum number of generations: M
Output global best solution (Optimal hyperparameters): H_{best}

begin
  generate an initial population of n chromosomes H_i $(i = 1, 2, 3, \ldots, n)$
  set generation counter $g = 0$
  while $g < M$ **do**
    train and evaluate the CNN model (population)
    new generation (retain the fittest individual)
    select the pair of chromosomes from the population based on fitness
    apply crossover operations on newly selected chromosome
    apply mutation on the offspring
    replace the old population with the new one
    $g = g + 1$
  end
  return H_{best}
end

3.6. Bootstrap aggregation ensembling

Ensemble learning is a widely used ML technique combining multiple base models to generate a robust and efficient prediction model (Al-Sarem et al., 2020). In this article, we selected the top five performing base models, including Generic CNN, Xception, Inception, Inseption-ResntV2, and EffcientNetV2L. The Bootstrap Aggregation Ensemble technique, also known as the bagging technique, is utilized to integrate the output of these base models. This technique improves the performance and accuracy of the models by dealing with variance trade-offs and reducing the variance of a prediction model. Bagging avoids overfitting data and can efficiently deal with higher dimensional data. The process of bagging ensembling is elaborated in Algorithm 2.

Algorithm 2: Bootstrap Aggregation Ensemble Algorithm

Input dataset $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$
base learning algorithms: $\mathcal{L}$
number of iterations: T

begin
  for $i = 1, 2, 3, \ldots, T$:
    D_i = Bootstrap (D) %Generate bootstrap sample from D
    $b_i = \mathcal{L}(D_i)$ % Train a base learner b_i from the bootstrap sample
  end
end
Output $E(x) = \text{argmax}_{y \in Y} \sum_{i=0}^{T} I(y = h_i(x))$

4. Experiments and performance evaluation

This section presents the experimentation methodology and performance evaluation of the proposed scheme. First, it describes the implementation platform, performance evaluation metrics, and utilized hyperparameters for model training. After that, a brief discussion of the results and comparing the proposed DTL with the state-of-the-art DTL-based IDSs is presented.

4.1. Implementation platform

The proposed DTL architecture is implemented, and performance is investigated using Keras libraries on the Google Colab Pro platform. The Google Colab Pro provides the Nvidia Tesla P100 graphic processing unit (GPU) with 16 GB graphic card memory and 25 GB RAM to ensure the smooth execution of DL algorithms. First, the image datasets are loaded from Google Drive through the ImageDataGenerator function in Keras. After that, 8 pre-trained CNN models are loaded into the Keras platform. During the training process, we fixed the number of epochs up to 25 and enabled the callback "save best model" based on the validation accuracy. During training, all the best-performing models are saved in Google Drive for future utilization. After the training, all the trained models are loaded from the drive, and performance is evaluated using the test dataset. The Google Colab notebooks of our implementation can be accessed from our GitHub repository on request and used for future endeavors.

4.2. Performance evaluation metrics

The performance of the proposed DTL framework is evaluated through several performance metrics, including accuracy, precision, recall, F1 score, Cohen Kappa score, and confusion matrices. These metrics are calculated according to the following equation.

Accuracy is the ratio of accurate predictions to all predictions.

$$Accuracy = \frac{True^{+} + True^{-}}{True^{+} + True^{-} + False^{+} + False^{-}} \quad (1)$$

Table 1
Optimal hyperparameters for the training of CNN models.

Model	Hyperparameters					
	Optimizer	Activation function	Dense units	Dropout	Fine tune layers	Epochs
Xception	Adamax	selu	128	0.2	128	14
VGG16	Adagrad	elu	128	0.5	8	11
VGG19	Adamax	relu	128	0.3	10	16
Inception	Adagrad	relu	128	0.5	45	19
InceptionResnetV2	Adamax	selu	128	0.3	451	15
EfficientNetB7	Adam	selu	128	0.4	288	15
EfficientNetV2L	Adam	selu	128	0.5	316	19

The precision is the ratio of true positives over the sum of false positives and true negatives.

$$Precision = \frac{True^{+}}{True^{+} + False^{+}} \tag{2}$$

The recall is the ratio of correctly predicted outcomes to all predictions.

$$Recall = \frac{True^{+}}{True^{+} + False^{-}} \tag{3}$$

The F1 score is defined as the harmonic mean of precision and recall.

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

Cohen's Kappa Statistic assesses the degree of agreement between two raters or judges who classify objects into mutually exclusive categories.

$$CohenKappaScore = \frac{P_o - P_e}{1 - P_e} \tag{5}$$

where

P_o : Relative observed agreement among raters

P_e : Hypothetical probability of chance agreement

The confusion matrix is used to determine the performance of the classification models for a given set of test data.

4.3. Hyperparameters for model's training

To ensure the optimum performance of the DL technique for the utilized dataset, the hyperparameters must be tuned before training. In the proposed framework, GA is incorporated for the HPO of each base learning model. In the proposed DTL, 8 CNN architectures with different hyperparameters are utilized. In GA, we defined the initial population of hyperparameters for each individual-based learning model. GA was executed for several generations and generated optimal hyperparameters for each model. All the CNN models are trained using these hyperparameters, and performance is investigated. The optimal hyperparameters for all base learning models are presented in Table 1.

4.4. Results and discussion

The experimentation and performance analysis was performed in two phases. In the first phase, a range of hyperparameters was supposed for all base learning models. All the models were trained using these hyperparameters. In the second phase, we incorporated GA to obtain the optimal hyperparameters. Again, all the base learning models were trained using optimal hyperparameters. For better visualization of results, we fixed the number of epochs to 25 in both scenarios. All the best base learning models are saved in Google Drive for future predictions. The best base learning models are saved on the criteria of maximum validation accuracy. First, we evaluate each model's prediction performance for non-optimized scenarios. First, all the trained models and test datasets are loaded from Google Drive. After that, the prediction results are obtained for each model in terms of accuracy, precision, recall, F1, CK score, and confusion matrix. The prediction results of all models without HPO are summarized in Table 2. In this phase, the performance of generic CNN and Xception models was almost similar. Both of these models attained 92.10% accuracies and successfully classified 8 types of cyberattacks. The performance score of VGG16, InceptionResnetV2, and EfficientNetV2L was less than 90%. The EfficientNetB7 was the third-best-performing model that attained an attack detection accuracy of 93.36%. The next stage is to integrate the output of some best-performing models. Here we selected the five best-performing models based on their highest attack detection accuracies. These models include Generic CNN, Xception, VGG19, Inception, and EfficientNetB7. The five best-performing models in the non-optimized scenario are compared in bar graph Fig. 5. The outputs of the best-performing models are integrated using the Bootstrap Aggregation Ensemble technique. This technique is very useful in improving the accuracy of the models by dealing with variance trade-offs and reducing the variance of a prediction model. The final ensemble model's attack detection accuracy is attained as 97.17%. The other performance metrics of final IDS precision, recall, F1, and CK score are 97.33%, 97.17%, 96.86%, and 94.33%, respectively. The confusion matrix of the final ensemble model of the non-optimized scenario is presented in Fig. 7.

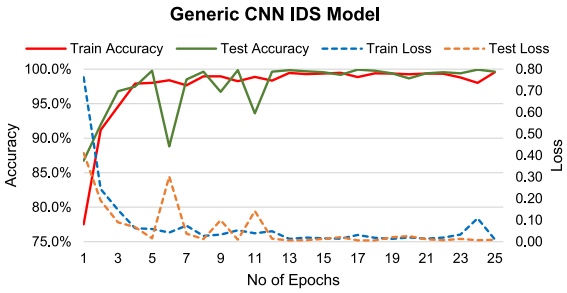
Table 2
Performance scores with non-optimized trained models.

Model	Performance metrics				
	Accuracy	Precision	Recall	F1 score	CK score
Generic CNN	0.9210	0.9020	0.9210	0.9079	0.8415
Xception	0.9210	0.8976	0.9210	0.9022	0.8417
VGG16	0.8979	0.8979	0.8979	0.8741	0.7925
VGG19	0.9769	0.9749	0.9769	0.9741	0.9537
Inception	0.9784	0.9711	0.9784	0.9736	0.9570
InceptionResnetV2	0.8837	0.8435	0.8837	0.8522	0.7665
EfficientNetB7	0.9336	0.9225	0.9337	0.9167	0.8670
EfficientNetV2L	0.8845	0.8539	0.8845	0.8597	0.8597

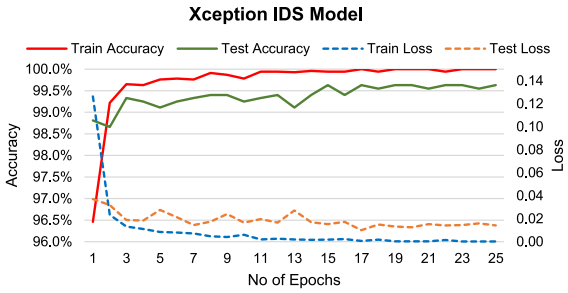
Table 3
Performance scores with optimized trained models.

Model	Performance metrics				
	Accuracy	Precision	Recall	F1 score	CK score
Generic CNN	0.9970	0.9970	0.9970	0.9969	0.9940
Xception	0.9977	0.9979	0.9977	0.9977	0.9955
VGG16	0.9955	0.9956	0.9955	0.9954	0.9910
VGG19	0.9925	0.9934	0.9925	0.9921	0.9850
Inception	0.9977	0.9979	0.9977	0.9978	0.9955
InceptionResnetV2	0.9985	0.9988	0.9985	0.9985	0.9970
EfficientNetB7	0.9821	0.9822	0.9821	0.9821	0.9641
EfficientNetV2L	0.9978	0.9979	0.9978	0.9977	0.9955

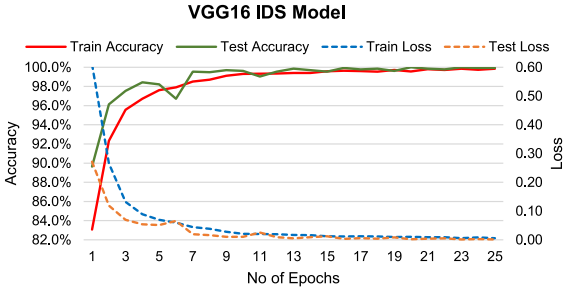
In the second phase of experimentation, we obtained the optimal hyperparameters proposed by GA. All the base learning models were trained on these parameters. The training and testing accuracies and losses for the optimized scenario are presented in Fig. 4. The prediction results of all optimized models are summarized in Table 3. The performance of generic CNN, InceptionResnetV2, and EfficientNetV2L was similar. These models obtained an attack detection accuracy of 99.93% and accurately classified 13 classes with 100% accuracy. Only 6% of samples of one class were misclassified. The second model was



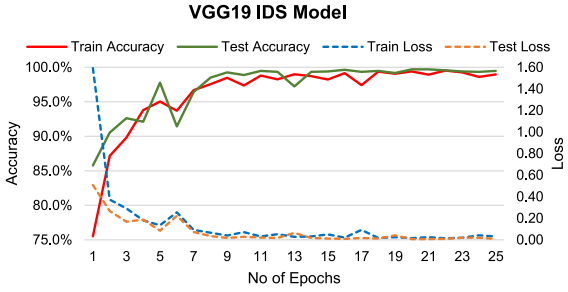
(a) Performance of CNN



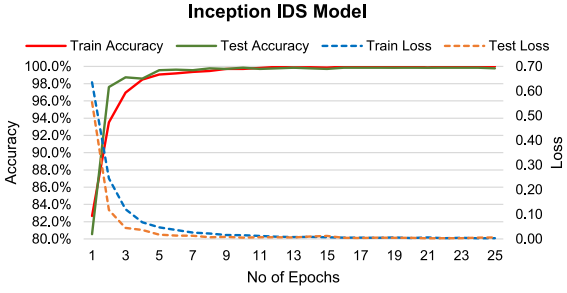
(b) Performance of Xception



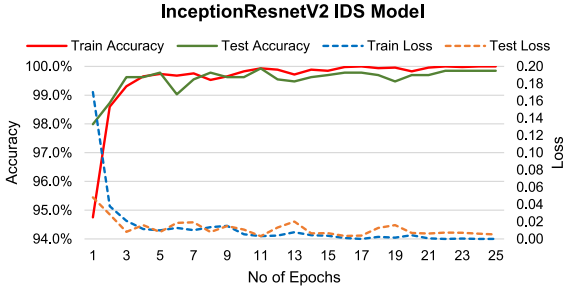
(c) Performance of VGG16



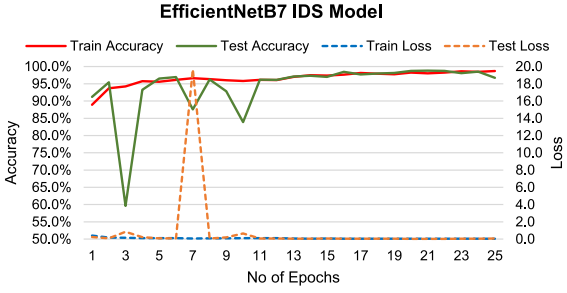
(d) Performance of VGG19



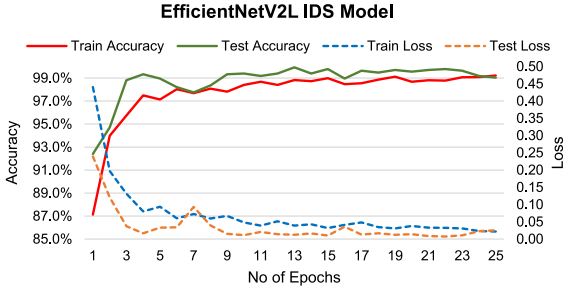
(e) Performance of Inception



(f) Performance of InceptionResnetV2



(g) Performance of EfficientNetB7



(h) Performance of EfficientNetV2L

Fig. 4. Training results of optimized CNN models.

Xception which attained an attack detection accuracy of 99.63% and accurately classified 12 classes of the Edge-IIoTset dataset. The third and fourth models are VGG16 and VGG19, respectively. The attack detection accuracies for VGG16 and VGG19 are recorded as 99.55% and 99.70%, respectively. The Inception model was the fourth best-performing model, with an accuracy of 99.85%. The attack detection accuracy of EfficientNetB7 was recorded as 98.851%, which is lower than all other models. Same as phase 1, the output of the five best-performing models was integrated through the Bootstrap Aggregation Ensemble technique. Here five best-performing models are Generic CNN, VGG19, Inception, InceptionResnetV2, and EfficientNetV2L. The five best-performing models for the optimized scenario are compared in bar graph Fig. 6. The final ensemble model attained an attack detection accuracy of 100%. The other performance metrics of final IDS precision, recall, F1, and CK score are also 100%. The proposed scheme accurately classified 14 classes with 100% accuracy. The confusion matrix of the final optimized model is presented in Fig. 8. The performance comparison of the final ensemble models is depicted in bar graph Fig. 9.

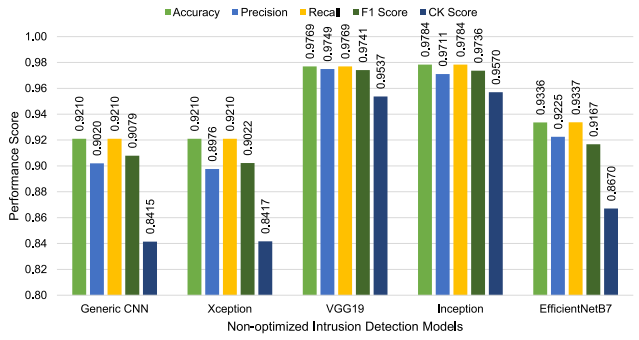


Fig. 5. Performance comparison of five best-performing models without HPO.

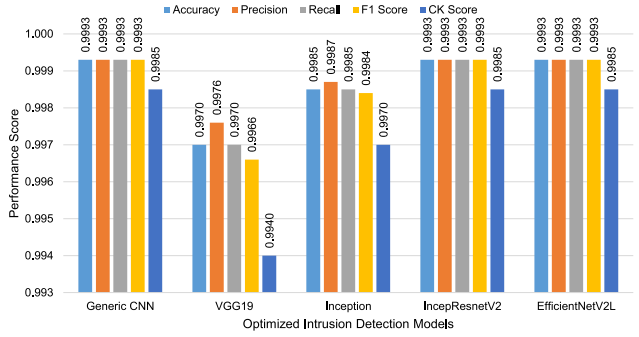


Fig. 6. Performance comparison of five best-performing models with HPO.

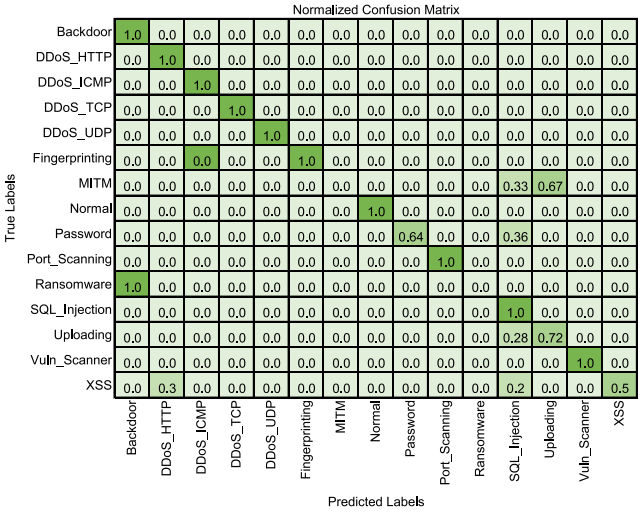


Fig. 7. Confusion matrix of the proposed DTL scheme without HPO.

4.5. Performance comparison with related DTL-based IDSs

To evaluate the effectiveness of the proposed DTL scheme, we compared its performance with some of the latest TL-based intrusion detection schemes. The DTL is not deeply explored for cybersecurity applications in the existing literature. Most researchers evaluated their proposed schemes through old-generation datasets containing limited classes of cyberattacks. Therefore, these IDS can predict a limited number of cyberattacks. In the proposed framework, we utilized the latest cybersecurity dataset, the Edge-IIoTset dataset. It was published in 2022 and is one of the more realistic and comprehensive datasets for benchmarking ML/DL-based intrusion detection schemes. The proposed DTL with Edge-IIoTset can predict 14 categories of cyberattacks. The second important comparison factor is that the existing DTL studies

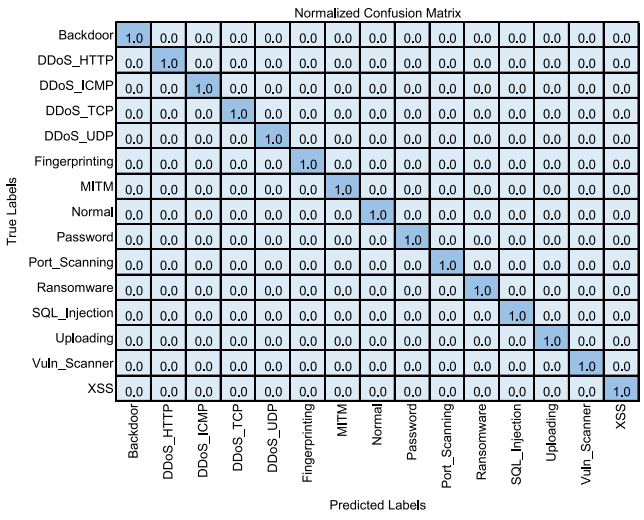


Fig. 8. Confusion matrix of the proposed DTL scheme with HPO.

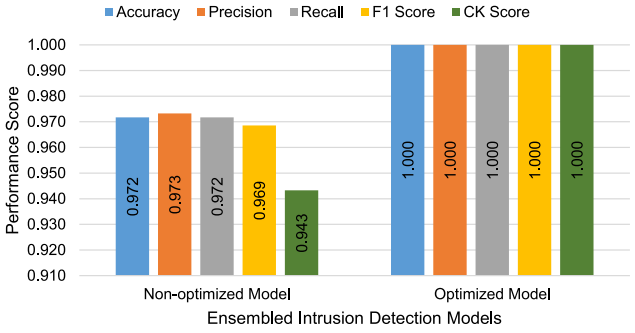


Fig. 9. Performance comparison of the final IDS models.

did not discuss the hyperparameter tuning for the training of their models. We incorporated a well-known meta-heuristic algorithm, GA, in the proposed scheme for hyperparameter optimization. The GA generates the optimal range of model parameters that ensure the best performance of each base learning model. The outcome of the proposed framework is its attack detection accuracy, proving its superior performance over other DTL-based schemes (see Table 4).

5. Conclusion

This article proposes an efficient and optimized framework for IIoT intrusion detection. The proposed scheme employs DTL, hyperparameter tuning, and ensemble learning. We used seven highly efficient CNN architectures, including Xception, VGG16, VGG19, Inception, InceptionResnetV2, EfficientNetB7, and EfficientNetV2L. To train these architectures, Edge-IIoTset was used as the most comprehensive and latest cybersecurity dataset. Moreover, GA is incorporated to ensure that CNN models perform well when trained with optimal parameters. To generate the final intrusion detection model, the outputs of the five best models were combined via a bootstrap aggregation ensemble algorithm. In order to assess the effectiveness of the proposed IDS, several performance metrics were defined. The proposed scheme outperformed various state-of-the-art intrusion detection systems in terms of attack detection accuracy. Looking forward, our future research will focus on further refining our scheme to develop a lightweight DTL that can be deployed in edge-enabled IIoT systems. As a commitment to fostering collaborative research, we have provided the complete source code and transformed dataset for other researchers to utilize and contribute to the development of DTL-based IDSs in the future.

Table 4
Performance comparison of the proposed framework with related transfer learning-based IDSs.

Reference	Proposed IDS scheme	Dataset	Predicted attacks	Hyperparameter optimization	Attack detection accuracy
Li et al. (2021)	TL	AWID	6	No	92%
Mehedi et al. (2022)	DTL	ToN_IoT	9	No	87%
Gou et al. (2009)	Distributed TL	Kdd Cup 99	4	No	97.3%
Mehedi et al. (2021)	DTL	Personally Generated Dataset	3	Yes	98.10%
Bierbrauer et al. (2023)	TL	UNSW-NB15, CICIDS2017	No Multiclass Evaluation	No	96.89%
Xu et al. (2021)	TL	CTU-13	No Multiclass Evaluation	No	93.01%
Singh et al. (2021)	DTL	ISCXVPN2016	No Multiclass Evaluation	No	96%
Abosata et al. (2023)	TL	RPL-IIoT dataset	No Multiclass Evaluation	No	85.52%
Yan et al. (2023)	TL	CICIDS2017, NSL-KDD	7	Yes	99.85%, 99.53%
Proposed Scheme	DTL	Edge-IIoTset	14	Yes	100%

CRediT authorship contribution statement

Shahid Latif: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Wadii Boulila:** Conceptualization, Software, Formal analysis,Investigation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition. **Anis Koubaa:** Formal analysis, Resources, Project administration, Supervision, Funding acquisition. **Zhuo Zou:** Conceptualization, Resources, Writing – review & editing, Visualization. **Jawad Ahmad:** Conceptualization, Validation, Writing – review & editing, Visualization.

Declaration of competing interest

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

Data availability

https://github.com/shahidlatif126/DTL-IDS-An-Optimized-Intrusion-Detection-Framework-using-DTL-and-GA.

Acknowledgment

The authors would like to thank Prince Sultan University for their support.

References

Abosata, Nasr, Al-Rubaye, Saba, Inalhan, Gokhan, 2023. Customised intrusion detection for an industrial IoT heterogeneous network based on machine learning algorithms called FTL-CID. Sensors 23 (1), 321.

Al-Sarem, Mohammed, Saeed, Faisal, Alsaeedi, Abdullah, Boulila, Wadii, Al-Hadhrami, Tawfik, 2020. Ensemble methods for instance-based arabic language authorship attribution. IEEE Access 8, 17331–17345.

Alzahem, Ayyub, Boulila, Wadii, Driss, Maha, Koubaa, Anis, Almomani, Iman, 2022. Towards optimizing malware detection: An approach based on generative adversarial networks and transformers. In: Computational Collective Intelligence: 14th International Conference, ICCCI 2022, Hammamet, Tunisia, September 28–30, 2022, Proceedings. Springer, pp. 598–610.

Bierbrauer, David A., De Lucia, Michael J., Reddy, Krishna, Maxwell, Paul, Bastian, Nathaniel D., 2023. Transfer learning for raw network traffic detection. Expert Syst. Appl. 211, 118641.

Chollet, François, 2017. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1251–1258.

Driss, Maha, Hasan, Daniah, Boulila, Wadii, Ahmad, Jawad, 2021. Microservices in IoT security: current solutions, research challenges, and future directions. Procedia Comput. Sci. 192, 2385–2395.

Ferrag, Mohamed Amine, Friha, Othmane, Hamouda, Djallel, Maglaras, Leandros, 2022a. Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications: Centralized and federated learning. https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot.

Ferrag, Mohamed Amine, Friha, Othmane, Hamouda, Djallel, Maglaras, Leandros, Janicke, Helge, 2022b. Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. IEEE Access 10, 40281–40306.

Gou, Shuiping, Wang, Yuqin, Jiao, Licheng, Feng, Jing, Yao, Yao, 2009. Distributed transfer network learning based intrusion detection. In: 2009 IEEE International Symposium on Parallel and Distributed Processing with Applications. IEEE, pp. 511–515.

Khan, Muhammad Almas, Khan Khattk, Muazzam A., Latif, Shahid, Shah, Awais Aziz, Ur Rehman, Mujeeb, Boulila, Wadii, Driss, Maha, Ahmad, Jawad, 2022. Voting classifier-based intrusion detection for iot networks. In: Advances on Smart and Soft Computing: Proceedings of ICACIn 2021. Springer, pp. 313–328.

Kim, Tae-Young, Cho, Sung-Bae, 2019. Particle swarm optimization-based CNN-LSTM networks for forecasting energy consumption. In: 2019 IEEE Congress on Evolutionary Computation. CEC, IEEE, pp. 1510–1516.

Latif, Shahid, e Huma, Zil, Jamal, Sajjad Shaukat, Ahmed, Fawad, Ahmad, Jawad, Zahid, Adnan, Dashtipour, Kia, Aftab, Muhammad Umar, Ahmad, Muhammad, Abbasi, Qammer Hussain, 2021. Intrusion detection framework for the internet of things using a dense random neural network. IEEE Trans. Ind. Inform.

Lee, Sang-Woong, Mohammadi, Mokhtar, Rashidi, Shima, Rahmani, Amir Masoud, Masdari, Mohammad, Hosseinzadeh, Mehdi, et al., 2021. Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review. J. Netw. Comput. Appl. 187, 103111.

Li, Xinghua, Hu, Zhongyuan, Xu, Mengfan, Wang, Yunwei, Ma, Jianfeng, 2021. Transfer learning based intrusion detection scheme for internet of vehicles. Inform. Sci. 547, 119–135.

Lokman, Siti-Farhana, Othman, Abu Talib, Bakar, Muhamad Husaini Abu, Musa, Shahrulniza, 2020. The impact of different feature scaling methods on intrusion detection for in-vehicle controller area network (CAN). In: Advances in Cyber Security: First International Conference. ACeS 2019, Penang, Malaysia, July 30–August 1, 2019, Revised Selected Papers 1, Springer, pp. 195–205.

Mehedi, Sk Tanzir, Anwar, Adnan, Rahman, Ziaur, Ahmed, Kawsar, 2021. Deep transfer learning based intrusion detection system for electric vehicular networks. Sensors 21 (14), 4736.

Mehedi, Sk Tanzir, Anwar, Adnan, Rahman, Ziaur, Ahmed, Kawsar, Rafiqul, Islam, 2022. Dependable intrusion detection system for IoT: A deep transfer learning-based approach. IEEE Trans. Ind. Inform..

Morid, Mohammad Amin, Borjali, Alireza, Del Fiol, Guilherme, 2021. A scoping review of transfer learning research on medical image analysis using ImageNet. Comput. Biol. Med. 128, 104115.

Moustafa, Nour, Hu, Jiankun, Slay, Jill, 2019. A holistic review of network anomaly detection systems: A comprehensive survey. J. Netw. Comput. Appl. 128, 33–55.

Natesha, B.V., Guddeti, Ram Mohana Reddy, 2021. Adopting elitism-based Genetic Algorithm for minimizing multi-objective problems of IoT service placement in fog computing environment. J. Netw. Comput. Appl. 178, 102972.

Shao, Siyu, McAleer, Stephen, Yan, Ruqiang, Baldi, Pierre, 2018. Highly accurate machine fault diagnosis using deep transfer learning. IEEE Trans. Ind. Inform. 15 (4), 2446–2455.

Simonyan, Karen, Zisserman, Andrew, 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Singh, Deepak, Shukla, Anurag, Sajwan, Mohit, 2021. Deep transfer learning framework for the identification of malicious activities to combat cyberattack. Future Gener. Comput. Syst. 125, 687–697.

Sisinni, Emiliano, Saifullah, Abusayeed, Han, Song, Jennehag, Ulf, Gidlund, Mikael, 2018. Industrial internet of things: Challenges, opportunities, and directions. IEEE Trans. Ind. Inform. 14 (11), 4724–4734.

Song, Hyun Min, Woo, Jiyoung, Kim, Huy Kang, 2020. In-vehicle network intrusion detection using deep convolutional neural network. Veh. Commun. 21, 100198.

Srinivas, Mandavilli, Patnaik, Lalit M., 1994. Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Trans. Syst. Man Cybern. 24 (4), 656–667.

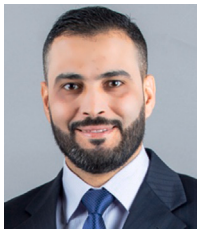
Szegedy, Christian, Vanhoucke, Vincent, Ioffe, Sergey, Shlens, Jon, Wojna, Zbigniew, 2016. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2818–2826.

Tan, Mingxing, Le, Quoc, 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. PMLR, pp. 6105–6114.

Tan, Mingxing, Le, Quoc, 2021. Efficientnetv2: Smaller models and faster training. In: International Conference on Machine Learning. PMLR, pp. 10096–10106.

Victoria, A. Helen, Maragatham, G., 2021. Automatic tuning of hyperparameters using Bayesian optimization. Evol. Syst. 12 (1), 217–223.
Wu, Di, Guan, Qinghua, Fan, Zhe, Deng, Hanhui, Wu, Tao, 2022. Automl with parallel genetic algorithm for fast hyperparameters optimization in efficient IoT time series prediction. IEEE Trans. Ind. Inform.
Xu, Mengfan, Li, Xinghua, Wang, Yunwei, Luo, Bin, Guo, Jingjing, 2021. Privacy-preserving multisource transfer learning in intrusion detection system. Trans. Emerg. Telecommun. Technol. 32 (5), e3957.
Yan, Fengru, Zhang, Guanghua, Zhang, Dongwen, Sun, Xinghua, Hou, Botao, Yu, Naiwen, 2023. TL-CNN-IDS: transfer learning-based intrusion detection system using convolutional neural network. J. Supercomput. 1–23.
Yang, Li, Shami, Abdallah, 2022. A transfer learning and optimized CNN based intrusion detection system for internet of vehicles. In: ICC 2022 - IEEE International Conference on Communications. pp. 2774–2779.
Yu, Tong, Zhu, Hong, 2020. Hyper-parameter optimization: A review of algorithms and applications. arXiv preprint arXiv:2003.05689.
Zingg, David W., Nemec, Marian, Pulliam, Thomas H., 2008. A comparative evaluation of genetic and gradient-based algorithms applied to aerodynamic optimization. Eur. J. Comput. Mech. (R. Eur. Méc. Numér.) 17 (1–2), 103–126.

Shahid Latif received the B.Sc. and M.Sc. degrees in Electrical Engineering from the HITEC University Taxila, Pakistan, in 2013 and 2018, respectively. He is pursuing a Doctoral degree at the School of Information Science and Technology, Fudan University Shanghai, China. He served as a Lecturer in the Department of Electrical Engineering, HITEC University Taxila, Pakistan, from 2015 to 2019. During his teaching career, he has supervised several projects in Electronics, Embedded systems, Control Systems, and the Internet of Things. In addition to his academic pursuits, he also gained practical experience as a cybersecurity researcher at Edinburgh Napier University, UK, and Prince Sultan University, Saudi Arabia.

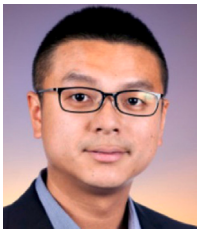


Wadii Boulila received the B.Eng. degree (1st Class Honours with distinction) in computer science from the Aviation School of Borj El Amri in 2005, the M.Sc. degree in computer science from the National School of Computer Science (ENSI), University of Manouba, Tunisia, in 2007, and the Ph.D. degree in computer science jointly from ENSI and Télécom Bretagne, University of Rennes 1, France, in 2012. He is currently an Associate Professor of computer science at Prince Sultan University, Saudi Arabia, and also a Senior Researcher with the RIOTU Laboratory at Prince Sultan University. Additionally, he is a Senior Researcher with the RIADI Laboratory, University of Manouba, and was previously a Senior Research Fellow with the ITI Department at the University of Rennes 1, France. Wadii received the Young Researcher in Computer Science Award for the year 2021 from Beit El-Hikma, the Best Researcher Award from the University of Manouba in Tunisia for the year 2021, and the Most Cited Researcher Award at the University of Manouba for the year 2022. He has participated in numerous research and industrially-funded projects. His primary research interests include data science, computer vision, big data analytics, deep learning, cybersecurity, artificial intelligence, and uncertainty modeling. He has served as the chair, reviewer, and TPC member for many leading international conferences and journals. His work has gained global recognition, and he has been nominated as one of the top 2 scientists in his field by Stanford University. Wadii Boulila is an IEEE Senior Member, an ACM member, and a Senior Fellow of the Higher Education Academy (SFHEA) in the U.K.

Anis Koubaa stands as a Computer Scientist, particularly noted for his pioneering work in Unmanned Aerial Systems and AI. Currently a distinguished Full Professor, Aide to the President of Research Governance, and the Director of Research and Initiatives at Prince Sultan University, his work spans a horizontal research scope, intersecting with multiple disciplines, including Unmanned Systems, Generative AI, Large Language Models, and the Internet-of-Things.

A significant portion of Professor Koubaa's expertise concentrates on Unmanned Aerial Systems and Drones. He pioneered a fleet management system for drones utilizing 4G/5G networks, accompanied by an Unmanned Traffic Management for airspace regulation. He also introduced surveillance drones with edge computing for deep learning-driven detection and tracking. Additionally, he developed drones for agricultural spraying and aerial image analytics, including traffic and road damage assessment.

With a bibliography including over 20 edited books with Springer and recognition as one of the top 2 Leadership Award. Professor Koubaa is a Senior Fellow of the UK's Higher Education Academy (HEA).



Zhuo Zou received the Ph.D. degree in electronic and computer systems from the KTH Royal Institute of Technology (KTH), Sweden, in 2012. Currently, he is with Fudan University, Shanghai, as a Full Professor, where he is conducting research on intelligent chips and systems for AIoT. Prior to joining Fudan, he was the Assistant Director and a Project Leader with the VINN iPack Excellence Center, KTH. He was an Adjunct Professor and a Docent with the University of Turku, Finland. His current research interests include low-power circuits, energy-efficient SoC, neuromorphic computing and their applications in AIoT, and autonomous systems. He is the Vice Chair of IFIP WG-8.12.

Jawad Ahmad (SMIEEE) is a highly experienced teacher with a decade of teaching and research experience in prestigious institutes. He has taught at renowned institutions such as Edinburgh Napier University (UK), Glasgow Caledonian University (UK) and Hongik University (South Korea) etc. He has also served as a supervisor for several Ph.D., M.Sc., and undergraduate students, providing guidance and support for their dissertations. He has published in renowned journals including IEEE Transactions, ACM Transactions, Elsevier and Springer with over 150 research papers and 4000 citations. For the past three years, his name has appeared on the list of the world's top 2 endorsed by Stanford University, USA). To date, he has secured research and funding grants totaling £195K in the UK and Norway as a Principal Investigator (PI) and a Co-Investigator (Co-I). In terms of academic achievements, he has earned a Gold medal for his outstanding performance in MS and a Bronze medal for his achievements in BS.