DTL-IDS: Deep Transfer Learning-based Intrusion Detection System in 5G Networks

Behnam Farzaneh*, Nashid Shahriar*, Abu Hena Al Muktadir†, Md. Shamim Towhid*
*Department of Computer Science
*University of Regina, Canada, †Saskatchewan Polytechnic, Canada
{behnamfarzaneh, nashid.shahriar, mty754}@uregina.ca, {muktadir.uec}@gmail.com

***Abstract*—In the complex landscape of modern networks, the necessity of Intrusion Detection System (IDS) has become paramount. An IDS is a crucial cybersecurity tool that plays a pivotal role in safeguarding networks against a wide array of threats and attacks. The application of deep learning models for intrusion detection is becoming popular among research communities due to its success in many other domains. However, deep learning models require a significant amount of labeled data to achieve effective training. Obtaining labeled data for intrusion detection can be challenging and costly. To address it, Deep Transfer Learning (DTL) can be employed. This research introduces an innovative traffic classification method tailored for 5G networks. The approach leverages deep transfer learning by utilizing pre-trained models and fine-tuning them. We evaluate several deep-learning models in a transfer learning setting. The Inception model being identified as the top-performing model shows an improvement of approximately 10% in terms of F1-score between IDS-based DTL and the same scheme without DTL.**

***Index Terms*—5G Networks, Intrusion Detection, Deep Transfer Learning**

I. Introduction

In modern networks such as 5G network, the challenge of intrusion detection has become increasingly complex due to the rapid increase of devices, the exponential growth of data traffic, and the dynamic nature of network architectures. Traditional Intrusion Detection Systems (IDS) often struggle to keep pace with the diverse and evolving attack techniques that target these networks. This challenge is exacerbated by the scarcity of labeled data specific to 5G environments, hindering the development of accurate detection models.

Transfer learning (TL) [1] is a machine learning approach where a model trained on one domain (called source domain) is re-purposed to perform a different but related task (called target domain). Instead of training a model from scratch for the target domain, TL leverages the knowledge and representations learned from the source domain to enhance the performance on the target domain. TL is particularly useful in situations where the target domain has limited data and features learned from the source domain are applicable to the target domain.

TL enables the transfer of learned features, patterns, and representations from the source domain to the target domain, aiding in the detection of novel and sophisticated attacks. TL also accelerates model convergence and reduces the resource-intensive process of building an IDS from scratch for 5G networks. By capitalizing on TL techniques, intrusion detection in modern networks like 5G can be more effective, adaptable, and responsive to emerging threats, ultimately enhancing security posture of these advanced communication infrastructures.

The goal of this paper is to implement and put into practice an effective intrusion detection approach based on TL, knowledge transfer, and model improvement. With scarce and imbalanced datasets, we assess the detection rate and accuracy for well-known and new Distributed Denial of Service (DDoS) attacks in 5G networks. In intrusion detection, two distinct but related datasets that include both benign traffic flows and attack traffic flows in 5G networks are taken into consideration.

Our dataset[1], which contains a large amount of network traffic generated in a 5G lab testbed, is chosen as the source domain. This dataset is an extended version of work [2]. The 5G-NIDD dataset [3], which has a sparse amount of annotated 5G network traffic pertaining to current and modern cyberattacks, is chosen as the target domain.

The main contributions of this paper are as follows:

- We collect six million traffic flows (attack and benign) from a 5G testbed based on our dataset and use that as a source domain in TL settings. The network has two slices in our testbed and the traffic flows are collected from both slices.
- We evaluate different deep learning algorithms as classifiers to detect attacks. For this purpose, we create the base model according to our dataset and apply a Deep Transfer Learning (DTL) approach for each algorithm on the target dataset. We show that the TL approach can enhance the performance of deep learning models in our experiments.

The rest of this paper is organized as follows. Section II discusses the related work on TL-based intrusion detection solutions. Section III provides the background used in our paper. Section IV introduces the proposed TL-based intrusion detection approach. Section V describes the usage of the approach and discusses the performance evaluation results. Section VI concludes the paper.

II. Related Work

In this section, we discuss some previous works related to TL from the literature. As TL is a well-studied area of deep learning in other domains such as Computer Vision and Natural Language Processing, there are a plethora of studies in the literature. However, the application of TL in a 5G network

[1]https://gitlab.com/behnam1394/dtl-ids-5g-dataset-2023

978-3-903176-59-1 ©2023 IFIP

for intrusion detection is not well studied. Hence, this paper studies the application of TL in the context of IDS in a 5G network.

The paper [4] proposes a unique defense method against evasion attacks including Fast Gradient Sign Method, Projected Gradient Descent, Carlini & Wagner, and DeepFool on network-based IDS. In order to improve communication between IDS and adversarial detectors, a TL approach was employed. To test defense's effectiveness against unseen evasion techniques, simulations of zero-day attack scenarios were run. The results indicate how this fusion strategy improves detection in a parallel IDS architecture, highlighting the effectiveness of the presented defense in comparison to other strategies.

The study [5] applies TL to address the dynamic network traffic and changing attack landscape in the Internet of Vehicles (IoV). Based on the accessibility of labeled data from the IoV cloud, two model updating strategies are suggested. To ensure reliable model updating, the first method uses cloud-assisted updates with less labeled data. When cloud-provided labels are not available, the second option, referred to as local updates, uses pseudo-labeled data instead. The effectiveness of these strategies has been tested using wireless network intrusion detection data, with results showing at least a 23% improvement in accuracy over more conventional techniques.

The paper [6] introduces the IDS-INT technique, which uses imbalanced network traffic to identify various attack types. IDS-INT analyzes imbalanced data to find feature correlations and network representations using TL. With the use of contextual anchors for precise feature mapping, data on network interactions is gathered to describe attacks thoroughly. The SMOTE approach improves minority attack comprehension while balancing network traffic. While a CNN-LSTM model detects attack types by learning deep features, a CNN model extracts deep features from balanced network data. IDS-INT outperforms baseline approaches in performance evaluation using well-known datasets, obtaining 99% precision, 100% recall, 99% F1-score, and 99.21% accuracy.

The efficacy of TL for detecting zero-day attacks in IoT networks with limited and imbalanced datasets is examined in [7]. This study uses the BoT-IoT dataset as the source domain for knowledge acquisition before applying it to the UNSW-NB15 dataset as the target domain. The development of a unique IDS that combines knowledge transfer and model improvement shows significant detection accuracy for both known and new cyberattack families. By applying the framework to a different dataset after training on one, its efficacy is confirmed. The results show that the proposed method provides great accuracy and very low false positive rates.

The paper [8] proposes a solution for traffic classification in 5G IoT networks, addressing challenges associated with limited datasets and computing resources. The method uses DTL, utilizing weight initialization and neural network fine-tuning to facilitate knowledge transfer from a source domain to a target domain. The results show that, when using only 10% of the USTC labeled dataset, the accuracy for LeNet-5, BiT, and EfficientNet-B0 models surpasses 95.47%, 96.22%, and 96.40%, respectively. These accuracies are closely matched to the results achieved with the full training data, which are 98.65%, 99.41%, and 98.68%, respectively.

In addition to the significant dataset and DTL approach, our proposed method uniquely studies the performance improvement of deep learning algorithms within a multi-slice 5G network, providing insights into the adaptability of these models across various network scenarios.

III. Background

A. Intrusion Detection

A breach in computer or network security is called an intrusion, and it frequently includes malicious packets being transmitted by attackers in an effort to steal or manipulate sensitive data. In order to detect possible security breaches, IDS collects and analyzes network traffic, security logs, and audit data from key nodes. The data collecting modules for gathering necessary data, the analysis modules that use machine learning and deep learning approaches to predict attacks, and the numerous detection methods with responses ranging from warnings to corrective actions are all part of the common framework of IDSs [1].

B. Classifier Algorithms

- **BiLSTM:** An enhanced variation of the Long Short-Term Memory (LSTM) algorithm is the Bidirectional LSTM (BiLSTM) [9]. In order to manage long-term dependencies and avoid the vanishing gradient issue that affects basic Recurrent Neural Networks (RNNs), LSTM implements the idea of memory cells in one or more hidden layers. In a standard LSTM, information only passes from the past to the future, but in BiLSTM, the structure includes both forward and backward LSTM layers, and the output layer processes the inputs from both levels concurrently [9].
- **CNN-based Algorithms:** CNN is a well-known deep learning architecture that is incredibly effective at performing image processing tasks. Three different types of layers are often found in a CNN: convolutional layers, pooling layers, and fully-connected layers. The feature patterns of data retrieved by convolution operations in a convolutional layer. Local correlations can be used in pooling layers to minimize data complexity without affecting crucial information and prevent over-fitting. All features are connected through fully connected layers, which also provide the output [10]. CNNs are the foundation of the two DTL solutions created in this paper.
  Two advanced CNN-based algorithms including ResNet and Inception are utilized to train base learners on network traffic data in our experiments. Microsoft Research introduced ResNet in 2015 [11]. The vanishing gradients issue in deep networks, which can cause underfitting during training, is addressed by this model. Using identity shortcut connections, the pre-trained ResNet model is employed as a meaningful residual extractor from raw data in

place of features. Using other layer combinations, refined versions of ResNet, such as ResNet-34, and ResNet-50, have been developed. Furthermore, Google created the deep CNN architecture known as GoogLeNet, commonly referred to as Inception-v1, for the ImageNet large-scale visual recognition challenge. It is frequently employed for image classification problems and has produced cutting-edge outcomes on a number of benchmarks [1].

C. Transfer Learning (TL)

Transfer learning is applying previously learned information, such as features and weights, to a new related domain. This includes transferring models developed on one dataset to another dataset. Consider a domain D, represented as $D = \{x, P(X)\}$, characterized by feature space x and marginal probability $P(X)$ for sample data point X. Label space Y and an objective function n, denoted probabilistically as $P(\gamma, X)$, make up a task T. In this case, a domain D establishes a task T as, $T = \{\gamma, P(Y|X)\} = \{\gamma, n\}$, where $Y = \{y_1, ..., y_n\}$ and $y_i \in \gamma$. So, a source domain D_s, associated source task T_s, target domain D_T, and target task T_T may all be used to define transfer learning. To ensure transfer learning, we put together the target conditional probability distribution $P(Y_T|X_T)$, which is integrated into D_T with information from D_S and T_S (when $D_S \neq D_T$ or $T_S \neq T_T$) [12].

DTL includes a variety of methods: Domain adaptation, which modifies various feature distributions and feature spaces between the source and target domains to improve the performance of the target learner; Multiple tasks within a domain are simultaneously learnt through a process known as "multi-task learning", regardless of the source or target designation; One-shot learning, which uses a small number of examples to categorize more instances, and Zero-shot learning, which uses no class instances and is independent of labeled data but requires additional training data to comprehend unobserved information [1].

IV. DTL-IDS: TL-Based IDS in 5G Networks

This paper presents an intrusion detection approach for DDoS attacks in 5G networks that are built on DTL. The approach is divided into two phases: the first is an initial training phase on the source domain, and the second is a phase of applying TL to the target domain. Both phases employ the same BiLSTM and CNN-based algorithms as their base learning components. Our approach involves keeping the convolutional base unchanged and utilizing its output to serve as input for the classifier.

We perform the following steps in our experiments:

A. Preprocessing

As part of the preprocessing step, we assigned labels to different types of DDoS attacks in both source and target datasets. This process involves categorizing and tagging the network traffic or instance in both datasets to show the specific kind of DDoS attack each instance represents. In our experiments, we used eight features in total that include 'Flow Duration', 'Fwd Pkt Len Std', 'ACK Flag Cnt', 'Protocol', 'Tot Fwd Pkts', 'Tot Bwd Pkts', 'TotLen Fwd Pkts', and 'TotLen Bwd Pkts'. These features are selected by following [2]. The same set of features are used in both source and target datasets. We train the models for two classes: attack and benign.

B. Base Model Construction

The BiLSTM and CNN-based models were utilized to build the base models using our dataset as the source domain. The model will perform better if the domains for which the TL may be employed are more comparable since inadequate data instances will be dealt with and training time will be cut down.

In this phase, we have different steps as follows:

- Train on the source dataset from scratch and save the model as a base model. These models are used for TL approach on the target dataset.
- We also train the same models from scratch on the target dataset in order to compare with the TL approach to see improvement between IDS-based TL and without TL.

C. Layers Freezing Method and Transfer Learning Phase

Layers freezing, in which we freeze the learned parameters of some of the pre-trained model's layers that have the generic features and only update (Fine-Tune) the parameters of the other layers that have the specific high-level features, is one of the methods utilized to enhance model performance and shorten training time in TL. Our experiments show the effectiveness of freezing all layers in the base model. In this case, we load the trained base model on the target dataset and freeze all layers by setting "trainable = False". During the training process, freezing includes limiting the update of weights inside a specified layer. The convolutional base undergoes freezing in this situation, resulting in its weights remaining fixed and untrainable throughout subsequent training phases. This measure is taken to protect important knowledge during later training phases. Our method includes the freezing of the convolutional base within the trained base model, allowing its outputs to act as inputs for the classifier in the pursuit of TL. For this purpose, we removed the last three layers from the base model to get only the convolutional base and added more layers. After that, we applied TL, and created a new model on top of output layers from the base model and then trained our new model on the target dataset, and finally saved the model.

D. Datasets

- **Our Dataset:** In this paper, we used our dataset as a source dataset that is based on network slicing (two slices are considered) and it includes benign and attack traffic. To generate benign traffic, we access 500 popular websites using a Python script. These websites include Streaming YouTube and live videos, Downloading, copying, removing, and pasting files between user equipment and server, ICMP ping, and SSH. Furthermore, our dataset contains about six million instances collected through different days. In addition, this dataset has 84 network traffic features and nine DDoS types of attacks: UDP flood, TCP syn, TCP push, TCP ack, TCP fin, TCP rst, TCP urg, TCP xmas,

TABLE I: Number of instances in each dataset

Our Dataset		5G-NIDD Dataset	
Attack Type	**Instances**	**Attack Type**	**Instances**
UDP flood	381073	UDP Flood	194946
TCP YMas	524288	SYN Flood	7566
TCP XMas	524288	GoldenEye	72499
TCP Urg	470757	ICMP flood	2
TCP SYN	405774	Slowloris	8669
TCP SRT	524288	UDP scan	33
TCP Push	524288	SYN scan	75
TCP FIN	916050	TCP Connect	189
TCP ACK	499729	TorShammer	31686
Benign	967567	Benign	15212

and TCP ymas. The time considered for capturing each kind of attack is two minutes. All attacks are done based on slice 1 and slice 2 with using "d" and "i" parameters in hping3 tool (i= 120 and d= 350). All packet captures of real-time network traffic are converted to CSV files by the CICFlowMeter traffic generator.

- **5G-NIDD Dataset:** This dataset is published by the authors of [3]. The data are collected from a fully functional 5G test network. Two types of attack (DoS and port scans) are performed during data collection. The DoS attacks are ICMP flood, UDP flood, SYN flood, HTTP flood, and Slowrate DoS. The authors perform three types of scan attack during the data collection. These attacks are SYN scan, TCP connect scan, and UDP scan. 5G-NIDD contains both application layer and transport layer attacks whereas, our dataset contains only transport layer attacks. The types of attacks in 5G-NIDD are different than our dataset, which justifies the choice of 5G-NIDD as the target dataset in transfer learning settings.

Table I shows the number of benign and attack traffic instances in both our dataset and the 5G-NIDD dataset. A measure for calculating the disparity between two distributions is the Maximum Mean Discrepancy (MMD) [13]. We compute the MMD score, which contrasts the distribution of the source and target datasets. The distributions are the same if the MMD score is 0, and different if the MMD value is larger than 0. We calculate the MMD score on the two selected datasets in our experiments. The MMD score is 0.2605 which illustrates how dissimilar the two distributions are in both datasets.

V. Evaluation Results

We carefully regulated many parameters in order to provide a rigorous and consistent experimental foundation and guarantee the comparability and reliability of our assessment results. Using a consistent random number seed across both datasets is an important step in this process. This careful process reduced the entry of needless randomness and variances in our evaluation results. Additionally, we strategically kept the same number of samples in each dataset's training and test sets to avoid further randomness in our evaluation results.

The number of training and test samples from the source and target datasets were kept in exactly the same proportion. Additionally, the essential issue of dataset balance was addressed by the design of our experimental setting. Given the importance of balanced datasets in generating trustworthy conclusions, we selected benign and attack traffic samples to preserve equilibrium. This required that one million samples of benign traffic and an equal number of samples of attack traffic (including all kinds of DDoS attacks) be carefully chosen from both classes. Notably, we conducted the experiments five times and took the average. The method of repeating the experiments multiple times allowed us to generate more comprehensive results. By taking the average of the outcomes from these five repetitions, we were able to arrive at a final set of results. After that, we selected the best result of our dataset out of five iterations in order to create the base model and then use it in the TL approach.

We implemented all algorithms using Keras running as a Python library on TensorFlow using Google Collaboratory cloud servers and Compute Canada [14] and a laptop with an 11th Generation Intel Core i7 processor and 16GB RAM. Both datasets include 8 features alongside the label, donating to the modeling process. With a binary classification task, the number of classes stands at 2. For training and validation, a test split of 0.2 (80% training data) and a validation split of 0.2 are used. The training process is run 200 epochs, augmented with early stopping mechanisms. The optimizer of choice is Adam, associated with a learning rate set at 1e-5, donating to the model's convergence and optimization.

In our experiments, an empirical evaluation was conducted to assess the performance of BiLSTM and CNN-based algorithms using both our dataset and the 5G-NIDD dataset. For this purpose, we used the well-known metrics namely, Accuracy, Recall, Precision, and F-1 score to evaluate the performance of our proposed method. Table II shows the average performance comparison in terms of the mentioned metrics for all algorithms.

Subsequently, the application of TL was investigated to ascertain its impact on algorithm performance based on 5G-NIDD dataset. Notably, three distinct performances in Figure 1 were obtained, each corresponding to the metrics' evaluations across the three different stages: our dataset, the 5G-NIDD dataset, and the 5G-NIDD dataset after applying TL. Upon careful investigation, results consistently showed that the algorithmic performance on the 5G-NIDD dataset with the TL approach yielded excellent results compared to the performance on 5G-NIDD dataset without TL.

As shown in figure 1, it becomes evident that the TL-based approach not only exhibits remarkable performance in contrast to the 5G-NIDD dataset but also showcases superior performance relative to our dataset. The outcomes highlight that BiLSTM on our dataset with 98.71%, Resnet on the 5G-NIDD dataset with 88.68%, and ResNet on the TL-based approach with 93.36% stand out with the highest levels of performance in terms of accuracy compared to others. However, Inception on the TL-based approach has the most improvement among all algorithms in terms of F1-score.

VI. Conclusion

We study the effect of deep transfer learning on network intrusion detection in this paper. DTL is a successful strat-

TABLE II: Performance Comparison

Model	Our Dataset				5G-NIDD Dataset				5G-NIDD Dataset with TL			
	Acc (%)	Rec (%)	Pre (%)	F1 (%)	Acc (%)	Rec (%)	Pre (%)	F1 (%)	Acc (%)	Rec (%)	Pre (%)	F1 (%)
BiLSTM	98.71	98.18	99.25	98.71	84.47	80.86	91.15	85.60	**90.60**	**89.72**	**92.00**	**90.85**
CNN	98.08	96.72	99.53	98.10	86.74	86.62	87.30	86.96	**88.86**	**90.11**	**87.68**	**88.86**
ResNet	91.04	88.67	94.81	93.15	88.68	93.47	83.53	88.02	**93.36**	**94.56**	**92.18**	**93.34**
Inception	91.05	91.63	90.66	91.06	84.14	88.53	81.35	82.82	**91.82**	**89.25**	**95.35**	**92.19**

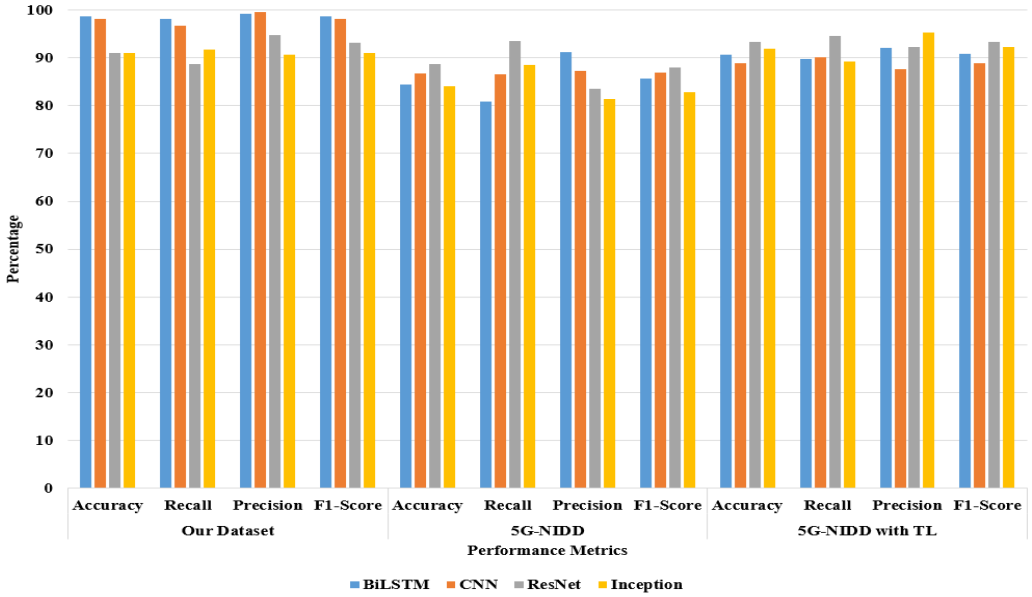

Fig. 1: Model Average performance (%) based on our dataset, 5G-NIDD dataset and 5G-NIDD dataset with TL

egy that can help IDSs better detect attacks and intrusions that conventional techniques can miss. Using BiLSTM and CNN-based algorithms, the model trains the base model by transferring the knowledge from our dataset to the 5G-NIDD dataset. Our experiment shows that the transfer learning approach can help the deep learning models achieve better performance. This finding highlights the method's flexibility and resilience, demonstrating its efficiency in handling the particular complexities of the 5G-NIDD dataset with transfer learning and boosting its ability. This empirical investigation clarifies how the algorithms behave in various situations, advancing our understanding of how well they work in practical situations.

ACKNOWLEDGEMENT

This work was supported in part by funding from the Innovation for Defence Excellence and Security (IDEaS) program from the Department of National Defence (DND).

REFERENCES

[1] Kheddar, H., Himeur, Y. and Awad, A.I., 2023. Deep Transfer Learning Applications in Intrusion Detection Systems: A Comprehensive Review. arXiv preprint arXiv:2304.10550.

[2] Khan, M.S., Farzaneh, B., Shahriar, N., Saha, N. and Boutaba, R., 2022, October. SliceSecure: Impact and Detection of DoS/DDoS Attacks on 5G Network Slices. In 2022 IEEE Future Networks World Forum (FNWF) (pp. 639-642). IEEE.

[3] Samarakoon, Sehan, et al. 5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network. arXiv preprint arXiv:2212.01298 (2022).

[4] Debicha, I., Bauwens, R., Debatty, T., Dricot, J.M., Kenaza, T. and Mees, W., 2023. TAD: Transfer learning-based multi-adversarial detection of evasion attacks against network intrusion detection systems. Future Generation Computer Systems, 138, pp.185-197.

[5] Li, X., Hu, Z., Xu, M., Wang, Y. and Ma, J., 2021. Transfer learning based intrusion detection scheme for Internet of vehicles. Information Sciences, 547, pp.119-135.

[6] Ullah, F., Ullah, S., Srivastava, G. and Lin, J.C.W., 2023. IDS-INT: Intrusion detection system using transformer-based transfer learning for imbalanced network traffic. Digital Communications and Networks.

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

[8] Guan, J., Cai, J., Bai, H. and You, I., 2021. Deep transfer learning-based network traffic classification for scarce dataset in 5G IoT systems. International Journal of Machine Learning and Cybernetics, 12(11), pp.3351-3365.

[9] Joseph, L.P., Deo, R.C., Prasad, R., Salcedo-Sanz, S., Raj, N. and Soar, J., 2023. Near real-time wind speed forecast model with bidirectional LSTM networks. Renewable Energy, 204, pp.39-58.

[10] Yang, L. and Shami, A., 2022, May. 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). IEEE.

[11] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[12] Otoum, Y., Wan, Y. and Nayak, A., 2022, May. Transfer learning-driven intrusion detection for Internet of Vehicles (IoV). In 2022 International Wireless Communications and Mobile Computing (IWCMC) (pp. 342-347). IEEE.

[13] Vu, L., Nguyen, Q.U., Nguyen, D.N., Hoang, D.T. and Dutkiewicz, E., 2020. Deep transfer learning for IoT attack detection. IEEE Access, 8, pp.107335-107344.

[14] Baldwin, S., 2012, February. Compute Canada: advancing computational research. In Journal of Physics: Conference Series (Vol. 341, No. 1, p. 012001). IOP Publishing.