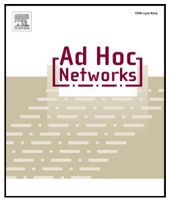
Ad Hoc Networks 166 (2025) 103645

Contents lists available at ScienceDirect

Ad Hoc Networks

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



Survey paper

PLLM-CS: Pre-trained Large Language Model (LLM) for cyber threat detection in satellite networks

Mohammed Hassanin [a,d,*], Marwa Keshk [b], Sara Salim [b], Majid Alsubaie [c], Dharmendra Sharma [c]

[a] University of South Australia (UniSA), SA, Australia
[b] University of New South Wales, Canberra, Australia
[c] University of Canberra, Canberra, Australia
[d] Fayoum University, Fayoum, Egypt

ARTICLE INFO

Keywords:
Large Language Models
LLMs
Intrusion detection
Transforms
Cyber threats
Satellites
Network security

ABSTRACT

Satellite networks are vital in facilitating communication services for various critical infrastructures. These networks can seamlessly integrate with a diverse array of systems. However, some of these systems are vulnerable due to the absence of effective intrusion detection systems, which can be attributed to limited research and the high costs associated with deploying, fine-tuning, monitoring, and responding to security breaches. To address these challenges, we propose a pre-trained Large Language Model for Cyber Security, for short PLLM-CS, which is a variant of pre-trained Transformers, which includes a specialized module for transforming network data into contextually suitable inputs. This transformation enables the proposed LLM to encode contextual information within the cyber data. To validate the efficacy of the proposed method, we conducted empirical experiments using two publicly available network datasets, UNSW_NB 15 and TON_IoT, both providing Internet of Things (IoT)-based traffic data. Our experiments demonstrate that proposed LLM method outperforms state-of-the-art techniques such as BiLSTM, GRU, and CNN. Notably, the PLLM-CS method achieves an outstanding accuracy level of 100% on the UNSW_NB 15 dataset, setting a new standard for benchmark performance in this domain.

1. Introduction

Recent advances in satellite communications have progressed the development of many end-user services, including the Internet of Things (IoT), Internet of Vehicles (IoV) and healthcare. These systems provide opportunities to develop physical applications such as smart cities and enterprise management systems. Satellites are complicated devices that provide many services and tasks. Satellites offer a means to extend wireless networks to unreachable places to terrestrial infrastructures [1]. One method of classifying satellites is the distance of their orbit from Earth relative to each other. These range from low earth orbit (LEO), which is closest to Earth, to geostationary earth orbit (GEO), which is the furthest from Earth. GEO orbits are slower with a wide orbital path, whilst LEO orbits are faster.

Satellites are launched for different purposes and missions. They are platforms for performing tasks based on their in-built equipment and sensors. Some are used to monitor and send images of the Earth to detect environmental changes. Others provide internet services to remote areas and facilitate applications, including healthcare emergencies and self-driving cars [2]. However, all have shared entities for providing basic services; for instance, data processing information collected from their sensors is an initial step for them. Their processes detect their orientations and positions, malfunctions, and diagnoses. Some actuators responsible for charging satellites using solar radiation have to be equipped with panels in their systems [3].

Satellite networks are vulnerable to cyber attacks, like many other systems, such as IoT and IoV. These threats become more severe when networks' data are dependent on physical devices such as satellites which, in turn, require more robust Intrusion Detection Systems (IDS); for example, according to the study in [4], 57% of IoT devices are exposed to severe attacks. Attacking satellites is more dangerous than attacking the IoT and other networks because they are vital for remote areas such as army units. Moreover, the cutting of connections among distant military units by satellite attacks can cause breakdowns in command and control, as demonstrated in the recent conflict between Ukraine and Russia [5]. If satellite networks are integrated with IoT devices, another gate is open for cyber attacks; for instance, a Mirai attack [6] based on a botnet along with a Distributed Denial of Service (DDoS) can exploit communications, such as data transport systems,

* Corresponding author.
E-mail address: mff00@fayoum.edu.eg (M. Hassanin).

https://doi.org/10.1016/j.adhoc.2024.103645
Received 7 May 2024; Received in revised form 17 August 2024; Accepted 1 September 2024
Available online 11 September 2024

to cause cyber threats. One of the main reasons behind these attacks is that these cyber threats are adaptive that need high reasoning of the machine learning. Hence, LLMs are the best variant of machine learning t fix such issues.

Satellite networks are composed mainly of ground stations, space segments and up-and-down links operated from the ground segments. The following attacks/adversaries can confront them. (1) a DoS based on overwhelming the target with dense traffic to prevent legitimate users from normal access [7]. It causes inaccessibility to resources and/or, ultimately, failure of a service. As reported in [8], there is steady growth in the number of DoS attacks, sizes, frequencies and complexities. A DoS has various types, such as ICMP, UDP, SYN and HTTP flood. Although different approaches use these types, all result in the final inaccessibility of the targeted system.

A DoS is a multidisciplinary attack that can be used in all network connections, including satellite ones. (2) A Distributed DDoS attack harmonizes multiple DoS ones simultaneously to invade a single target system., It has more resistance and complexity than a DoS one due to its multiple instances with a single aim. As a result, it is more severe and disruptive because it can access more resources, and it is unrealistic to shut down all infected systems simultaneously. Such attacks are related to terrestrial networks, as detailed in [9–11]. However, they are still disruptive to SSNs because LSN networks can be attacked globally, and SSNs provide global information. Also, they have other features, including low latency, limited numbers of clients and sparse connectivity between ground and space segments which make an attacker's task easy.

To address the above-mentioned issues, Smart Satellite Networks (SSN) that integrate satellite systems, IoT over network communications, and Machine Learning paradigms have been proposed [12,13]. One direction, especially for military space systems, is encryption-based methods [14,15]. In [16], elliptic curve cryptography is proposed for securing satellite communications. However, these methods secure only the physical layer and IDSs for SSNs have been investigated on a limited scale. In one study [17], a Convolutional Neural Network (CNN) and Long Short-term/Temporary Memory (LSTM) models are used inside a federated learning architecture to detect adversaries.

In this paper, we propose using self-attention modules [18] to build a robust system capable of detecting adversaries with high levels of accuracy. The main contribution is the development of a robust attention-based IDS, namely, a Pre-Trained Large Language Model for Cyber Security Defence (PLLM-CS), to determine the presence of advanced adversaries in SSNs. To our knowledge, this is the first study to propose a transformer-based method for detecting satellite adversaries. Although it is considered a centralized approach compared to a distributed one, it still outperforms the baselines. Also, it performs better than the benchmarks in detecting intrusions in any network data. It uses transformers at the model's core because of their capabilities to learn long-term contextual representations. Following are the contributions of this research.

- Developing a Pre-Trained Large Language Model for Cyber Security Defence method as an IDS for SSNss to determine highly advanced adversaries, including fuzzer, DoS and reconnaissance attacks.
- Illustrating that this method is a generic solution for any network by validating it on satellite and IoT network data.
- Providing extensive experiments using various datasets with different attacks and comparing traditional and deep learning models.
- Providing a new benchmark for IDSs on two publicly available datasets.

2. Related works

Satellite Network attacks Satellite Network Attacks: although satellite systems are similar to terrestrial ones *w.r.t* network data, they have different methodologies. As a DoS attack is popular in all types of networks, it is more serious when deployed in a satellite one because of its wide coverage and involvement of multiple technologies [19]. However, a DoS adversary can target multiple things in SSNs, including the connections between their nodes, in the same way as attacking classical networks using millions of bots and botnets. In Coremelt [20], inter-domain links are targeted by adversaries to cause congestion. A number (N) of bots and a botnet are used to generate legitimately similar flows to bypass their adversaries. However, these adversaries can then initiate flows that bypass the links in the target, causing congestion in any link. Crossfire [21] is similar to Coremelt [20] in its way of attacking. The links are congested and overloaded as the connections between a network's topology are hindered. Coremelt and Crossfire are difficult to alleviate because they are indistinguishable from legitimate traffic. Another attack that targets a satellite's up and down links is proposed in ICARUS [22]. It generates legitimate traffic to overload the communications of an LSN and is considered the easiest because of its low bandwidth. It causes congestion to an ISL with more traffic and combines multiple links to construct a more complicated form. In general, it creates more threats than the previous two attacks because some of their mitigations are not practically applicable to it.

Satellite IDS Satellite IDS: Recently, SSNs have progressed significantly in covering rural areas and providing cheap services for industrial sectors, such as healthcare and the IoT and IoV, particularly in the absence of wireless networks. However, they require robust IDSs to maintain the continuity of these services. In [23], Zhenyu et al. proposed a new mechanism based on the design of distributed routing LEO SSNs. It uses classical Machine Learning (ML) methods to estimate the network traffic and then intelligent routing decisions are made to maintain the traffic without any overloads. Gunn et al. [24] proposed using LSTM to detect any anomalies in network data which, as a result, reduces the number of false alarms in satellite communication systems. Another study in [25] was proposed to monitor a spacecraft and maintain the health of its entire system. Dictionary learning and sparse representation detect intrusions in satellite network data. Cheng et al. [26] used LSTM to predict anomalies in SSNs and then define a system's health as the difference between its actual and expected parameters. Although machine learning methods have been investigated to detect anomalies in network traffic, LLMs have not received enough attention yet. Also, cyber attacks are very adaptive that yield different variations of the attacks regularly. The inference capability of LLMs make them the best option to handle such threats.

In [27], the authors proposed a method for detecting anomalies in satellite telemetry data. Firstly, the Deviation Divide Mean over Neighbors (DDMN) technique is used to detect intrusions in the data. Then, the LSTM learns deep features from the multivariate data. Finally, a Gaussian model is employed to detect intrusions in the LSTM's features. In the most recent study [28], Zeng et al. proposed CN-FALSTM, a data-driven technique for detecting anomalies in the telemetry data of a satellite. Its main objective is to mitigate the false positive rates. Likewise, Yun et al. [29] developed a model for predicting the voltage and current of a satellite in a low orbit. Most recently, Moustafa et al. [13] introduced a federated learning IDS based on LSTM to detect intrusions on satellite systems.

Transformers as IDS Since Transformers [18] have been proposed as an attention-based solution to different paradigms, such as vision and NLP, great progress has been made in all aspects of ML [30,31]. In [32], Tan et al. used an attention-based technique for real-time detection because of the time-slot capabilities of transformers. They compared bidirectional LSTM (BiLSTM) and Conditional Random Fields (CRFs) as baselines with their proposed method performing the best In a similar study [33], Wu proposed using a typical transformer design consisting

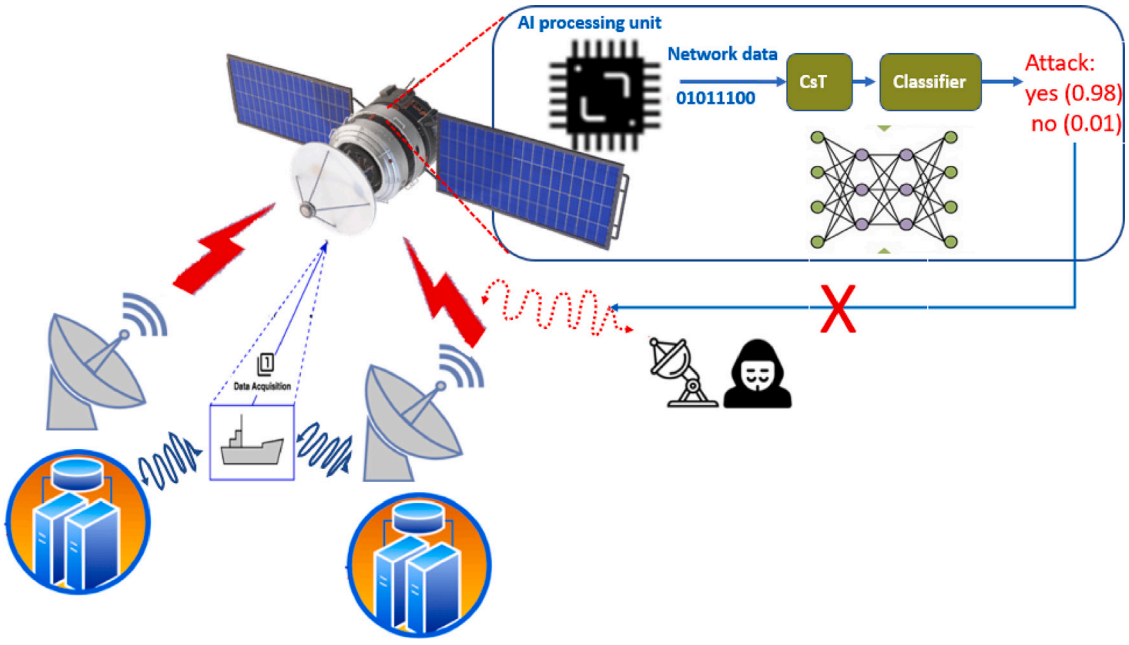


Fig. 1. The visualization of the proposed method, PLLM-CS, as an IDS with the satellite network. PLLM-CS will be an embedded AI-processing unit such as Nvidia Jetson, plugged into the satellite.

of positional encoding to identify tokens, encoders to learn low features and then self-attention modules to encode long-range relationships. In [34], Ghourabi used different structures of transformers to detect intrusions in network data. It benefited from the power of self-attention modules to consider the context of the input data to protect healthcare systems from cyberattacks. In a recent study, Luo et al. [35] introduced a modified fused architecture from a CNN and transformers to detect intrusions in network data which involved a proposed CNN-transformer NIDS and traffic spatiotemporal features. It uses softmax to encode the selection of soft features to improve the capability of the final model. Although these models are applied to different applications, such as network data and healthcare systems, no IDS for SSNs using the most recent technology of deep ML transformers has been investigated.

3. Method

The architecture of the proposed method is shown in Fig. 1.

3.1. Pre-processing

Initially, transformers aimed to help natural language-processing applications, such as text translations and summarizations [36]. In them, the text is a set of sentences, each of which is a set of words representing a token. However, cyber-security-network data are different because they are not sentences but multivariate series. This means that their contexts and long-range relationships are not used, limiting the power of transformers. To address this issue, we propose forming sentences from the multivariate series' tokens and then dividing them into tokens to learn the long-term relationships among them. Put, as each input feature is considered a word, putting these features together is how to form a sentence (see Fig. 2).

3.2. Pre-trained large language transformer model

Self-attention modules have improved the performance of ML tasks, including visual recognition [37], natural language processing, and multimodal ones. The most popular module of such self-attention architecture is Transformers [18]. Despite its great success, less attention is paid to cyber-security paradigms. Following previous studies in the literature, [18], the input is first fed into a preprocessing step at the beginning and then goes through transformer encoders stacked on top of each other.

The input corresponds to the multivariate sequence $\{\mathbf{x}_i \in \mathbb{R}^J | i = 1, \ldots, N\}$, where J is the total number of variables in the sequence and N is the total number. In the initial layer of the transformer, positional information is mixed with input patches as the identities of similar tokens, called patch embedding, and is calculated as:

$$Z_0 = [\mathbf{x}_1 E; \mathbf{x}_2 E; ..., \mathbf{x}_P E], \quad E \in \mathbb{R}^{(J)\times C}, Z_0 \in \mathbb{R}^{T\times C} \tag{1}$$

where C is the embedding dimension, and T is the number of patches.

The result of the previous step Z_0, patch embedding, is fed into the core step of Transformers, *i.e.,*the *self-attention*. It implicitly learns the dependencies between the various tokens by encoding the relationships between the three main matrices $Q, K, V \in \mathbb{R}^{T\times C}$.

The inner operation is a scaled dot-product between these three matrices to encode attention scores as follows:

$$A(Q, K, V) = \text{Softmax}(\frac{Q.K^T}{\sqrt{P}}).V \tag{2}$$

In the above equation, Q and K are replicated matrices from the input to be used inside softmax. A dot-product operation is performed for those matrices as a similarity or correlation measure. Then, softmax is applied to the output of the dot-product, which decides the attention scores from 0 to 1. However, these attention scores are discrete and disconnected from the inputs and might cause gradient vanishing. As a result, two more operations are provided to ensure stability in training [18] as follows: (1) scaling the output of the dot product by $\sqrt{m}$, which reduces the weight variance. (2) multiplying this scaled-dot product output by matrix V to preserve the spatiality of the input features.

Although this is the main operation of self-attention modules, it is applied through multi-head attention (MHA), which performs simultaneously with various representations. This MHA is achieved by concatenating all the heads:

$$MHA = Concat(A_i(.)).W, \quad i \in 1, \ldots, H \tag{3}$$

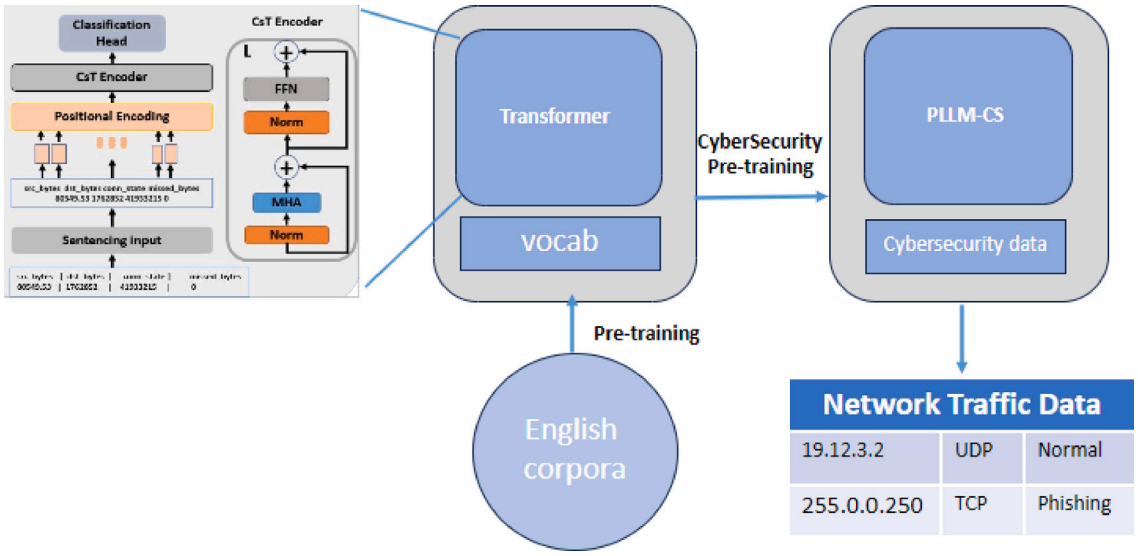


Fig. 2. Visual explanation for PLLM-CS components. Firstly, the input is forwarded to the sentencing step to facilitate context encoding by MHA. The main contribution is to handle binary datasets with LLMs, as well as fine-tuning the parameters to achieve the highest accuracy.

where W is the learned weight of each matrix, whereas H is the number of heads. MHA proves the ability to encode attention from different positions jointly.

The whole self-attention module is then stacked with more layers, including multi-layer perceptron (MLP) and layer normalization [38]. At each layer of the transformer l, the list of operations is performed in this sequence:

$$\begin{aligned} Z_l &= Mask(MHA(LN(Z_{l-1}))) + Z_{l-1}, \\ Z_l &= MLP(LN(Z_l)) + Z_l, \\ Z_l &= LN(Z_l), \\ where \quad & l = 1, 2, \ldots, L \end{aligned} \tag{4}$$

where $LN(.)$ refers to the layer normalization, l is the layer indicator and L is the number of layers on the transformer. In addition, MHA means multi-head attention for the input at layer l, whereas + refers to matrix summation. This set of operations can encode all the relationships amongst the input tokens by dividing the input into tokens and then applying these operations to temporally measure the correlations between tokens. Moreover, these operations do not depend on the previous batches of the input, which enables the model to encode the long-range relationships in parallel from multiple batches simultaneously and, as a result, improves the performance significantly.

In this implementation, we mask the self-attention modules to prevent cheating on the following tokens and prevent the attention mechanisms from leaking predictions based on the previous tokens.

3.3. Classification head

In this work, we consider binary classification IDSs to classify the input as an attack or not. Transformer encoders are stacked together as shown in Fig. 2. The output after L Transformer encoders has the shape $\mathbb{R}^{B\times T\times C}$ where B is the batch size, T is the token size and C is the channel size. Then, a fully connected layer is used to map it to $\mathbb{R}^{B\times 2}$, where 2 refers to binary classification.

3.4. Loss function

Binary cross entropy is the objective function to penalize the network in the case of wrong predictions as follows:

$$CE(y, p) = -\frac{1}{n}\sum_{i=1}^{n} y_i \log(p_i), \tag{5}$$

where y refers to the number of ground-truth labels, which is 2 in this study, p denotes the predicted probabilities,

4. Experiments

In this part, we discuss the evaluations of PLLM-CS on the network datasets, including UNSW-NB-15 [39] and TON_IoT [40]. In the absence of a public dataset for satellite security, these datasets are chosen because they include network data very similar to those of satellites. The settings for the experiments, baselines, datasets and evaluation metrics are provided at the beginning of this section. Then, visual comparisons with state-of-the-art methods are provided to highlight the significance of the PLLM-CS compared to the baselines.

4.1. Experimental setup

PLLM-CS implementation models are implemented on Pytorch, cuDNN, and CUDA-11 on their backends. To guarantee fair comparisons, the same settings are used for all baselines in the training stage. AdamW is the main optimizer with the same settings as Transformers [18]. The server that is used is Quadro GV100 with 32GB. 30 is the number of epochs, whereas the learning rate is $2e-5$. Eventually, the remaining settings follow Transformers. The models are used without any fine-tuning or transfer learning.

4.2. Datasets

UNSW-NB 15: provides network data for anomaly detection in several applications, including satellite security. It contains 49 network features, but we chose 13 of them. The chosen features measure the performance of the network flow, including port, IP, bytes, TTL, load, packets for both source and destination, and duration. It also contains normal data and attack data. Table 1 shows the details of this dataset.

TON_IoT: is an IoT dataset that contains telemetry data, network flow data, and operating systems logs. These data were collected for IoT and cybersecurity. It contains some common attacks, including backdoors, Denial of Service (DoS), injection, XSS, scanning, etc. It includes around 22 million records and 45 features. Out of 45 features, we choose eleven, as shown in Table 2.

(a) Training (b) Validation

Fig. 3. Visual comparisons between PLLM-CS, CNN, and BiLSTM on the training (left) and testing (right) phases on the UNSW NB-15 dataset. The proposed method's accuracy is higher than the baselines.

Table 1
Description of the network data on UNSW-NB 15.

Feature	Type	Description
srcip	nominal	The source IP
dstip	nominal	The destination IP
proto	nominal	The type of protocol
Sload	Float	The source BPS
Dload	Float	The destination BPS
Stime	Timestamp	The starting time
Ltime	Timestamp	The ending time
Spkts	Timestamp	The number of packets
label	Binary	The class of the record

4.3. Baselines

The significance of PLLM-CS is evaluated by comparing two cases. The first is traditional classifiers, including the Random Forest (RF), Extra Trees Classifier (ETC), Extreme Gradient Boosting (XGB), and Light Gradient Boosting Method (LGBM), while the second is trained-ML algorithms. The former advanced cyber-security branches during the last decade because of their simplicity and efficiency. However, they are not train-based algorithms, which affects their generalization factor. Also, they are more vulnerable to adversarial attacks than deep learning models. The latter is the preferred way of learning in recent decades, with models trained for a certain number of epochs to learn the patterns of the problem. We compare the PLLM-CS with a CNN, LSTM, BiLSTM, fully-connected NN(FNN) and Gated Recurrent Network (GRU).

4.4. Evaluation protocols

We use common evaluation metrics, including recall, precision, F measurement (F_1) and precision, to evaluate the proposed method, PLLM-CS. They are explained in detail as follows:

Accuracy: is one of the basic metrics for evaluating performance. The correct predictions are mainly referred to as T_p in contrast to wrong predictions T_n. Accuracy is the ratio of correct samples to the total:

$$Acc = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \tag{6}$$

Precision: This metric calculates the score of the true predicted samples T_p compared to the correct ones.

$$Prec = \frac{T_p}{T_p + F_p} \tag{7}$$

Recall: The recall metric is usually used in performance evaluation to calculate how many actual positives the model has predicted. It is important to detect the detection rates of the attacks.

$$Rec = \frac{T_p}{T_p + F_n} \tag{8}$$

F-measure (F_1) is a measure to balance the benefits of Precision and Recall. Compared to Accuracy, F_1 is important as a balance between precision and recall, particularly in the cases of uneven distributions for the classes where large numbers of negatives are present.

$$F - measure = 2 \times \frac{Prec \times Rec}{Prec + Rec} \tag{9}$$

False Negative Rate (FNR) as the name reveals, it is the ratio between false negatives and the total positives (false negatives and true positives)

$$FNR = \frac{F_n}{F_n + T_p} \tag{10}$$

Receiver Operating Characteristic (ROC) is a graphic criterion that plots the true positive rate (TPR) versus the false positive rate (FPR) with different threshold settings. The chart of ROC uses the following metrics on the x- and y-axis, respectively.

$$TPR = \frac{T_p}{T_p + F_n} \tag{11}$$

$$FPR = \frac{F_p}{F_p + T_n} \tag{12}$$

Area Under the ROC Curve (AUC) is the summation of the performance with all the threshold settings. It is a desirable measure because it is scale-invariant and classification-threshold invariant, which measures how well the predictions are ranked and how irrelevant they are to the classification's threshold.

Matthews Correlation Coefficient (MCC) is considered the best classification metric that summarizes the confusion matrix and considers the four metrics TP, TF, FP, and FN.

$$MCC = \frac{T_p \times T_n - F_p \times F_n}{\sqrt{(T_p + F_p)(T_p + F_n)(T_n + F_p)(T_n + F_n)}} \tag{13}$$

In the above settings, T_p refers to true positives, T_n represents true negatives, false positives are F_p, whereas false negatives are F_n.

Baselines settings For UNSW-NB 15, we chose non-trained models such as Random Forrest, ETC, XGB, and LGBM. Also, deep learning models include convolution neural network (CNN), long-short-term model (LSTM), Bidirectional LSTM (BiLSTM), fully connected neural network (FNN), and gated recurrent unit (GRU). For all the models, the number of epochs is 100, the batch size is 16, the feature size is 32, and the learning rate is 1×10^{-3}. The FNN model has three main linear layers and two non-linear layers (ReLU [41]), while the last is the classifier. CNN model has two main layers, each composed of 1D convolution, batch normalization [42], and ReLU, whereas the last layer is the classifier. The feature map size is 64 and 128, respectively. Recurrent models include two layers with a size of 64 for LSTM and GRU, whereas BiLSTM is 128. For the proposed model, the number of epochs is 10, and the number of Transformers blocks is 2. The dimension of the attention block is 32 with 4 heads. The loss function is cross entropy for all the variants, whereas the optimizer is AdamW [43].

Table 2
Description of the network data on TON_IoT.

Feature	Type	Description
time	time	time of logging data
date	date	date of logging data
motion status	number	motion sensor status (1 is on and 0 is off)
light status	boolean	whether the status of light is on or off
temperature	number	temperature sensor measurement data
pressure	number	pressure sensor reading
humidity	number	humidity sensor reading
sphone signal	boolean	status of door signal on phone
latitude	number	value of GPS tracker latitude
longitude	number	value of GPS tracker longitude
FC1 Read Input Register	number	a code for handling readings of input register
label	number	the class of record as attacked or normal

Table 3
Comparisons between PLLM-CS and both types of models, non-trained ML models and trained ML models on UNSW NB-15. The results show that PLLM-CS is outperforming both types of baselines.

	Model	Accuracy	Precision	Recall	F-measure	FNR	AUC	MCC
Non-trained ML models	RF	95.2	94.8	94.9	94.9	6.7	94.8	89.6
	ETC	95.1	94.6	94.8	94.7	6.8	94.6	89.4
	XGB	95.0	94.5	94.7	94.6	7.4	94.7	89.0
	LGBM	95.2	94.6	94.9	94.8	7.3	94.9	89.6
Trained ML models	CNN	83.5	86.9	78.1	80.2	6.8	78.2	64.5
	LSTM	70.6	67.9	66.7	67.1	39.2	66.7	34.6
	BiLSTM	77.2	75.5	74.5	74.9	29.8	74.5	50.0
	FNN	65.2	82.4	51.8	42.9	0	15.6	51.8
	GRU	74.3	72.2	71.7	71.2	34.7	71.7	43.9
	PLLM-CS	100	100	100	100	0.0	100	100

UNSW NB-15 Results In this part, we discuss the experimental results of the UNSW NB-15 dataset between the proposed method, PLLM-CS, and the baselines (RF, ETC, XGB, LGBM, CNN, LSTM, BiLSTM, FNN, and GRU). The results are illustrated in Table 3. Firstly, comparing PLLM-CS to non-trained machine learning models *i.e.,*traditional machine learning classifiers, the proposed method obtains better accuracy from the first few training epochs. Though these methods are simple in architecture and complexity, they provide high accuracy, up to 95.2%. The improvement amongst such models is very minor. For instance, the difference between RF and ETC is only 0.1%. Overall, PLLM-CS obtains better than all non-trained models with a large margin of 4.8%. Compared to deep learning models *e.g.*CNN, LSTM, BiLSTM, GRU, and FNN, PLLM-CS shows great significance. It obtains the highest accuracy 100% in all the metrics, while the second best is LSTM, with 93.0 accuracies. Though deep learning models are inferior to pre-deep learning models, trained models are generalizing better than non-trained ones. Overall, the proposed method outperforms both types with a considerable margin of 4.9% compared to the closest one. Diving into details, the metrics of the comparisons are accuracy, precision, recall, f-measure, AUC, and MCC. The proposed method showed significance in all the metrics. For example, in accuracy, it is the highest at 99.9. Also, F-measure, recall, precision, and MCC reported better results with PLLM-CS. It is worth noting that LSTM behaves better than CNN and FNN because of its ability to encode the context better than the input tokens. The reason behind this is encoding the context in the feature space. This is because of the presence of self-attention modules that are capable of learning long-range relationships.

The visual comparisons between the proposed methods and the previous deep learning models (*e.g.*CNN and BiLSTM) on TON_IoT are shown in Figs. 5 and 6. PLLM-CS shows significant validation accuracy with a large margin during the stability of the training. More precisely, Fig. 6 shows that the behavior of PLLM-CS in loss convergence is far better than the baselines. PLLM-CS is achieving higher performance than CNN and LSTM with better loss convergence.

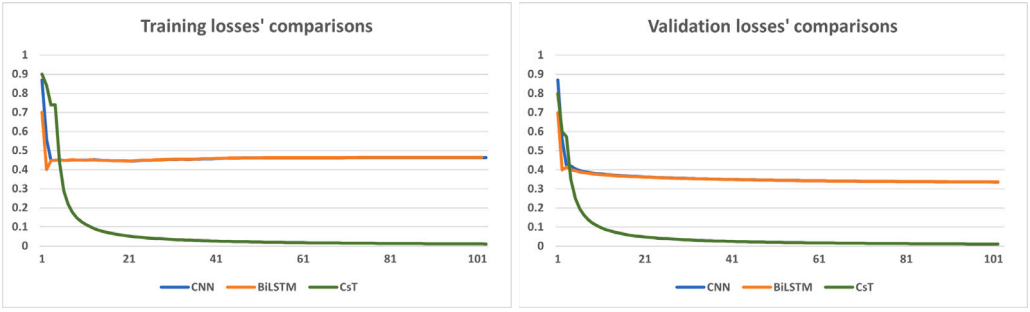
TON IOT Results: In this part, we discuss the experimental results on the TON_IoT dataset between the proposed method, PLLM-CS, and the baselines (RF, ETC, XGB, LGBM, CNN, LSTM, BiLSTM, FNN, and GRU). In Table 4, the results are illustrated. Firstly, PLLM-CS was compared to non-trained machine learning models. The proposed method obtained the highest accuracy, 100%, after a few training epochs. Though these methods are simple in architecture and complexity, they provide high accuracy, 100%. These models achieved a 0 False Negative Rate (FNR). Overall, PLLM-CS behaves better than non-trained models even though they achieved the full mark. This is because PLLM-CS is a trained-based algorithm that behaves better with novel examples and can generalize. Compared to deep learning models *e.g.*CNN, LSTM, BiLSTM, GRU, and FNN, PLLM-CS shows great significance. It obtains the highest accuracy 100% in all the metrics, while the second best is CNN with 98.27 accuracies. Though deep learning models are inferior to pre-deep learning models, trained models are generalizing better than non-trained ones. Overall, the proposed method outperforms both types. Diving into details, the metrics of the comparisons are accuracy, precision, recall, f-measure, AUC, and MCC. The proposed method showed significance in all the metrics. For example, in accuracy, it is the highest with 100%. Also, F-measure, recall, precision, and MCC reported better results with PLLM-CS. Again, it is proved that PLLM-CS behaves better because it can encode the context in the feature space.

The visual comparisons between the proposed methods and the previous deep learning models (*e.g.*CNN and LSTM) on TON_IoT are shown in Figs. 3 and 4. PLLM-CS shows significant validation accuracy with a large margin while the training stability. More precisely, Fig. 4 shows that the behavior of PLLM-CS in loss convergence is far better than the baselines. PLLM-CS is achieving higher performance than CNN and LSTM with better loss convergence.

Conclusively, PLLM-CS provides a great advantage in providing robustness for SSNs. This study provides Transformer-based IDSs for SSNs. As shown in the experiments section, PLLM-CS illustrates significant accuracy in two publicly available datasets, which include network

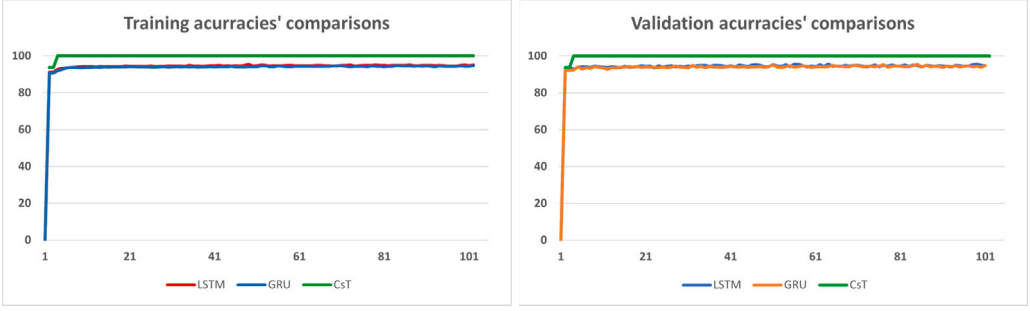
Table 4
Comparisons between PLLM-CS and both types of models, non-trained ML models and trained ML models on TON_IoT. The results show that PLLM-CS is outperforming both types of baselines.

	Model	Accuracy	Precision	Recall	F-measure	FNR	AUC	MCC
Non-trained ML models	RF	100.0	100.0	100.0	100.0	0	100.0	100.0
	ETC	100.0	100.0	100.0	100.0	0	100.0	100.0
	XGB	100.0	100.0	100.0	100.0	0	100.0	100.0
	LGBM	100.0	100.0	100.0	100.0	0	100.0	100.0
Trained ML models	CNN	98.27	98.35	97.9	98.12	1.8	97.9	96.8
	LSTM	94.6	94.7	94.1	94.1	4.1	94.1	88.2
	BiLSTM	94.5	93.9	94.5	94.2	3.3	94.5	88.5
	FNN	90.78	93.37	86.88	89.05	12.2	86.88	79.99
	GRU	94.7	93.5	95.2	94.2	1.5	95.3	88.8
	PLLM-CS	100.0	100.0	100.0	100.0	0	100.0	100.0



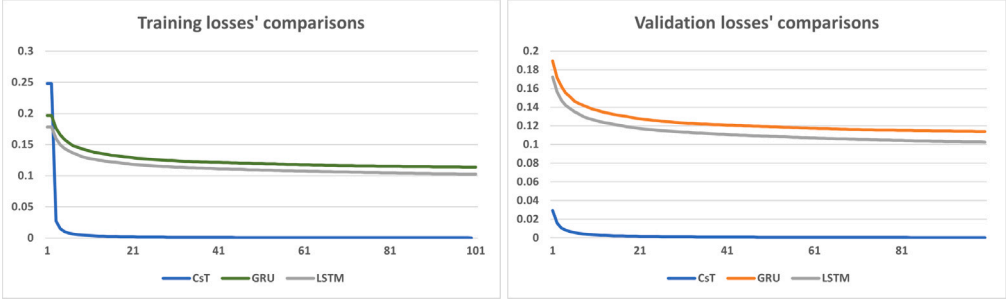
(a) Training (b) Validation

Fig. 4. Visual comparisons of losses for PLLM-CS, CNN, and BiLSTM on the training (left) and testing (right) phases on the UNSW NB-15 dataset. The proposed method shows stable convergence.



(a) Training (b) Validation

Fig. 5. Visual comparisons between PLLM-CS, GRU, and LSTM on the training (left) and testing (right) phases on the TON_IoT dataset. The proposed method's accuracy is higher than the baselines.



(a) Training (b) Validation

Fig. 6. Visual comparisons of losses for PLLM-CS, CNN, and BiLSTM on the training (left) and testing (right) phases on the TON_IoT dataset. The proposed method shows stable convergence.

data from diverse IoT domains. Also, PLLM-CS performance achieved the full mark of 100% without any fine-tuning or hyper-parameter optimization.

5. Conclusion and future work

In this study, a simple yet efficient intrusion detection model using contextual transformers, the PLLM-CS, to detect intrusions on the network data of SSNs is proposed. It adapts transformers to suit cybersecurity datasets by sentencing the input data, enabling them to encode long-term relationships. This is the first study to use transformers and attention-based models to detect intrusions on SSNs. The empirical results obtained from two real datasets, UNSW-NB 15 and TON_IoT, show the superiority of the proposed model over the baselines of RF, XGB, CNN, FNN, GRU, LSTM and BiLSTM. We conclude that this is because of its capability to encode contextual information using self-attention modules. However, as a satellite system does not have a proper dataset that mimics real data, the PLLM-CS's real-time network data used for testing are similar to satellite ones. Therefore, a future direction would be to develop a special dataset for SSNs and another to consider the constraint of the limited power inside them that requires efficient algorithms to increase their speeds and power consumption. Also, light versions of LLMs are required to fit the limited capacity of satellite resources.

CRediT authorship contribution statement

Mohammed Hassanin: Software, Methodology. **Marwa Keshk:** Writing – review & editing. **Sara Salim:** Writing – original draft. **Majid Alsubaie:** Writing – review & editing. **Dharmendra Sharma:** Writing – review & editing.

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

No data was used for the research described in the article.

References

[1] M. De Sanctis, E. Cianca, G. Araniti, I. Bisio, R. Prasad, Satellite communications supporting internet of remote things, IEEE Internet Things J. 3 (1) (2015) 113–123.

[2] C. Badue, R. Guidolini, R.V. Carneiro, P. Azevedo, V.B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T.M. Paixao, F. Mutz, et al., Self-driving cars: A survey, Expert Syst. Appl. 165 (2021) 113816.

[3] A. Colagrossi, M. Lavagna, A spacecraft attitude determination and control algorithm for solar arrays pointing leveraging sun angle and angular rates measurements, Algorithms 15 (2) (2022) 29.

[4] IoT adoption and its security risks have both grown, URL https://start.paloaltonetworks.com/unit-42-iot-threat-report.

[5] Russia downed satellite internet in Ukraine: Western officials, URL https://www.aljazeera.com/news/2022/5/10/russia-behind-cyberattack-against-internet-network-in-ukraine.

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

[7] S. Wankhede, D. Kshirsagar, DoS attack detection using machine learning and neural network, in: ICCUBEA, IEEE, 2018, pp. 1–5.

[8] Stats of cars, 2022, https://www.verisign.com/enIN/securityservices/ddosprotection/ddosreport/index.xhtml. (Online; accessed 27 April 2022).

[9] F. Lau, S.H. Rubin, M.H. Smith, L. Trajkovic, Distributed Denial of Service Attacks, vol. 3, IEEE, 2000, pp. 2275–2280.

[10] A.D. Wood, J.A. Stankovic, Denial of service in sensor networks, Computer 35 (10) (2002) 54–62.

[11] N. Borisov, G. Danezis, P. Mittal, P. Tabriz, Denial of service or denial of security? in: Proceedings of the 14th ACM Conference on Computer and Communications Security, 2007, pp. 92–102.

[12] K. Li, H. Zhou, Z. Tu, W. Wang, H. Zhang, Distributed network intrusion detection system in satellite-terrestrial integrated networks using federated learning, IEEE Access 8 (2020) 214852–214865.

[13] N. Moustafa, I.A. Khan, M. Hassanin, D. Ormrod, D. Pi, I. Razzak, J. Slay, DFSat: Deep federated learning for identifying cyber threats in IoT-based satellite networks, IEEE Trans. Ind. Inform. (2022).

[14] S. Jackson, J. Straub, S. Kerlin, Exploring a novel cryptographic solution for securing small satellite communications., Int. J. Netw. Secur. 20 (5) (2018) 988–997.

[15] M. O'Neill, E. O'Sullivan, G. McWilliams, M.-J. Saarinen, C. Moore, A. Khalid, J. Howe, R. Del Pino, M. Abdalla, F. Regazzoni, et al., Secure architectures of future emerging cryptography safecrypto, in: Proceedings of the ACM International Conference on Computing Frontiers, 2016, pp. 315–322.

[16] A. Ostad-Sharif, D. Abbasinezhad-Mood, M. Nikooghadam, Efficient utilization of elliptic curve cryptography in design of a three-factor authentication protocol for satellite communications, Comput. Commun. 147 (2019) 85–97.

[17] R. Zhao, Y. Yin, Y. Shi, Z. Xue, Intelligent intrusion detection based on federated learning aided long short-term memory, Phys. Commun. 42 (2020) 101157.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).

[19] H.R. Ibraheem, N.D. Zaki, M.I. Al-mashhadani, Anomaly detection in encrypted HTTPS traffic using machine learning: a comparative analysis of feature selection techniques, Mesopotamian J. Comput. Sci. 2022 (2022) 18–28.

[20] A. Studer, A. Perrig, The coremelt attack, in: European Symposium on Research in Computer Security, Springer, 2009, pp. 37–52.

[21] M.S. Kang, S.B. Lee, V.D. Gligor, The crossfire attack, in: 2013 IEEE Symposium on Security and Privacy, IEEE, 2013, pp. 127–141.

[22] G. Giuliari, T. Ciussani, A. Perrig, A. Singla, {ICARUS}: Attacking low earth orbit satellite networks, in: 2021 USENIX Annual Technical Conference, USENIX ATC 21, 2021, pp. 317–331.

[23] Z. Na, Z. Pan, X. Liu, Z. Deng, Z. Gao, Q. Guo, Distributed routing strategy based on machine learning for LEO satellite network, Wirel. Commun. Mob. Comput. 2018 (2018).

[24] L. Gunn, P. Smet, E. Arbon, M.D. McDonnell, Anomaly detection in satellite communications systems using lstm networks, in: 2018 Military Communications and Information Systems Conference, MilCIS, IEEE, 2018, pp. 1–6.

[25] B. Pilastre, L. Boussouf, S. d'Escrivan, J.-Y. Tourneret, Anomaly detection in mixed telemetry data using a sparse representation and dictionary learning, Signal Process. 168 (2020) 107320.

[26] F. Cheng, X. Guo, Y. Qi, J. Xu, W. Qiu, Z. Zhang, W. Zhang, N. Qi, Research on satellite power anomaly detection method based on LSTM, in: 2021 IEEE International Conference on Power Electronics, Computer Applications, ICPECA, IEEE, 2021, pp. 706–710.

[27] Y. Wang, J. Gong, J. Zhang, X. Han, A deep learning anomaly detection framework for satellite telemetry with fake anomalies, Int. J. Aerosp. Eng. 2022 (2022).

[28] Z. Zeng, G. Jin, C. Xu, S. Chen, Z. Zeng, L. Zhang, Satellite telemetry data anomaly detection using causal network and feature-attention-based LSTM, IEEE Trans. Instrum. Meas. 71 (2022) 1–21.

[29] S.-T. Yun, S.-H. Kong, Data-driven in-orbit current and voltage prediction using Bi-LSTM for LEO satellite lithium-ion battery SOC estimation, IEEE Trans. Aerosp. Electron. Syst. (2022).

[30] M. Hassanin, A. Khamiss, M. Bennamoun, F. Boussaid, I. Radwan, CrossFormer: Cross spatio-temporal transformer for 3D human pose estimation, 2022, arXiv preprint arXiv:2203.13387.

[31] M. Hassanin, S. Anwar, I. Radwan, F.S. Khan, A. Mian, Visual attention methods in deep learning: An in-depth survey, 2022, arXiv preprint arXiv:2204.07756.

[32] M. Tan, A. Iacovazzi, N.-M.M. Cheung, Y. Elovici, A neural attention model for real-time network intrusion detection, in: LCN, IEEE, 2019, pp. 291–299.

[33] Z. Wu, H. Zhang, P. Wang, Z. Sun, RTIDS: a robust transformer-based approach for intrusion detection system, IEEE Access (2022).

[34] A. Ghourabi, A security model based on LightGBM and transformer to protect healthcare systems from cyberattacks, IEEE Access 10 (2022) 48890–48903.

[35] S. Luo, Z. Zhao, Q. Hu, Y. Liu, A hierarchical CNN-transformer model for network intrusion detection, in: CAMMIC, vol. 12259, SPIE, 2022, pp. 853–860.

[36] M. Zaheer, G. Guruganesh, K.A. Dubey, J. Ainslie, C. Alberti, S. Ontanon, P. Pham, A. Ravula, Q. Wang, L. Yang, et al., Big bird: Transformers for longer sequences, Adv. Neural Inf. Process. Syst. 33 (2020) 17283–17297.

[37] M. Hassanin, S. Khan, M. Tahtali, A new localization objective for accurate fine-grained affordance segmentation under high-scale variations, IEEE Access 8 (2019) 28123–28132.

[38] J.L. Ba, J.R. Kiros, G.E. Hinton, Layer normalization, 2016, arXiv preprint arXiv:1607.06450.

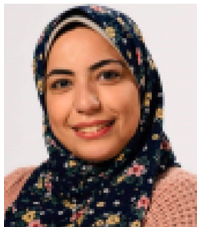
[39] N. Moustafa, J. Slay, UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set), in: MilCIS, IEEE, 2015, pp. 1–6.

[40] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, A. Anwar, TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems, IEEE Access 8 (2020) 165130–165150.

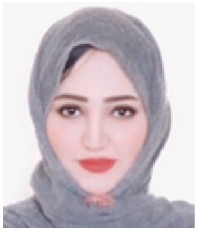
[41] A.F. Agarap, Deep learning using rectified linear units (relu), 2018, arXiv preprint arXiv:1803.08375.

[42] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: F. Bach, D. Blei (Eds.), Proceedings of the 32nd International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 37, PMLR, Lille, France, 2015, pp. 448–456.
[43] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, in: International Conference on Learning Representations, 2018.

Mohammed Hassanin is a Lecturer at the University of South Australia. He received his Ph.D. in Computer Science from the University of South Wales, Canberra and Master degree from Cairo University, Egypt. Before his Ph.D., he was a Computer Vision Researcher at a leading automotive construction industry warehouse. His research interests include computer vision, function understanding, deep learning, machine learning, and robotics.



Marwa Keshk is Lecturer in Cyber Security at the school of Professional studies, University of New South Wales (UNSW Canberra). She obtained her Ph.D. degree in Cyber Security and Privacy Preservation, and master degree in Evolutionary Computation in 2021 and 2017, respectively, from UNSW. She received the Bachelor degree in Computer Science in 2012 from the faculty of Computers and Information at Helwan University, Egypt. During my Ph.D., she was research candidate at Data61-CSIRO in Australia. Her areas of Interest include working in the intersection of Cyber Security and Artificial Intelligence for Privacy Preservation, Anomaly Detection and Threat Intelligence using Computational Intelligence, Machine Learning, Statistical Methods, and emerging technologies.



Sara Salim is a research associate in space systems at UNSW. She holds a Ph.D. in cybersecurity from UNSW Canberra (2023), a Bachelor's in Computer Science from Zagazig University, Egypt, and a Master's in Optimization and Operations Research Applications from Menoufia University, Egypt. Her research explores cybersecurity, AI, and device/network security, with interests in social networks, IoT, satellite communications, and space systems.

Majid Alsubaie works currently as a program manager at Saudi embassy in Canberra and academic fellow at University of Canberra. Through his experience, Majid blends theoretical knowledge and practical application approach in his daily tasks to benefit his employers. Majid holds a Ph.D. in information systems, master's degree in software engineering, and a Bachelor of computer engineering. Majid's research interest includes a multidisciplinary approach that encompasses the fields of Information Systems, Cyber Security Management, Agile Methodology, and Software Development.

Dharmendra Sharma is currently a Professor of Computer Science at the University of Canberra (UC). He is Chair of Faculty Board, Faculty of Science and Technology and had been the Chair of University Academic Board (2014–2019) and the Dean of the Faculty of Information Sciences and Engineering from 2007–2012 and as Head of School of the School of Information Sciences and Engineering from 2004–2007 at UC. He has assumed various senior leadership roles in universities for over twenty years and had been made a University Distinguished Professor by UC in 2012. Prof Sharma was bestowed Order of Australia (in AM division) in 2019 in recognition of his contribution to Higher Education and Computer Science.