On the Feasibility of Deep Learning in Sensor Network Intrusion Detection

Safa Otoum, *Student Member, IEEE*, Burak Kantarci, *Senior Member, IEEE*, and Hussein T. Mouftah, *Life Fellow, IEEE*

***Abstract*—In this letter, we present a comprehensive analysis of the use of machine and deep learning (DL) solutions for IDS systems in wireless sensor networks (WSNs). To accomplish this, we introduce restricted Boltzmann machine-based clustered IDS (RBC-IDS), a potential DL-based IDS methodology for monitoring critical infrastructures by WSNs. We study the performance of RBC-IDS, and compare it to the previously proposed adaptive machine learning-based IDS: the adaptively supervised and clustered hybrid IDS (ASCH-IDS). Numerical results show that RBC-IDS and ASCH-IDS achieve the same detection and accuracy rates, though the detection time of RBC-IDS is approximately twice that of ASCH-IDS.**

***Index Terms*—Wireless sensor network, cybersecurity, restricted Boltzmann machine, deep learning, machine learning, intrusion detection.**

I. Introduction

THE INTEGRATION of WSNs in critical applications has introduced security threats, such as jamming. Security susceptibilities can occur in either cyber or physical domains, including intrusions to communication links and sensor nodes. Intrusion Detection (ID) was introduced as an essential solution for network security, to deal with intrusive activities in communication networks and detect various intrusion attempts automatically [1].

Here, we present a detailed feasibility study of deep learning (DL)-based intrusion detection in the monitoring of critical infrastructures through sensor networks. Thus, our aim is to investigate the potential of deep learning as an alternative to robust machine learning (ML)-based intrusion detection systems. We consider our previously proposed Adaptively Supervised and Clustered Hybrid (ASCH-IDS) methodology [2] as a benchmark to assess the feasibility of a deep learning-based intrusion detection system. For a deep learning solution, we present a Restricted Boltzmann-based Clustered IDS (RBC-IDS) model for intrusion detection in WSN-based critical applications networks. We compare ASCH-IDS (i.e., ML-based) and RBC-IDS (i.e., DL-based) via simulations, and show that the accuracy of both approaches is above 99%, with the DL-based RBC-IDS detection rate slightly over 99%. However, training and estimating times of ASCH-IDS are ≈54% and 50% that of the RBC-IDS. Based on our findings, we propose that an ML-based IDS system is preferable to a DL-based IDS system under the circumstances for an exemplary case of WSN-based critical infrastructure monitoring.

Manuscript received September 21, 2018; revised November 16, 2018; accepted February 19, 2019. Date of publication February 26, 2019; date of current version May 23, 2019. This work was supported by the Natural Sciences and Engineering Research Council of Canada-DISCOVERY Program. The associate editor coordinating the review of this paper and approving it for publication was C. Wang. *(Corresponding author: Safa Otoum.)*

S. Otoum is with the School of Electrical Engineering and Computer Science, Faculty of Engineering, University of Ottawa, Ottawa, ON K1N 6N5, Canada (e-mail: sotoum@uottawa.ca).

B. Kantarci and H. T. Mouftah are with the University of Ottawa Faculty of Engineering, Ottawa, ON K1N 6N5, Canada (e-mail: burak.kantarci@uOttawa.ca; mouftah@uottawa.ca).

Digital Object Identifier 10.1109/LNET.2019.2901792

II. Background and Motivation

The essential design of any deep learning network requires using a Restricted Boltzmann Machine (RBM) as an unsupervised learning method [3]. Examples of this include the work done in [4] and [5]. Alom *et al.* [6] explored the capabilities of Deep Belief Networks (DBN) for detecting intruders through a series of experiments. DBN and SVMs have been introduced for intrusion detection classification purposes on the KDDCup99 dataset. With DBN as a feature selector and SVM as a classifier, the results showed a 92.84% accuracy rate [7]. Partial supervised learning approaches are presented in [8], Fiore *et al.* used real world data to evaluate their approach. A hybrid approach based on DBN and auto-encoder is shown in [9]. The auto-encoder method is used to decrease data dimensionality and extract the main features. After feature reduction, the DBN is applied to detect anomalous behaviour. The work in [10] employed the Restricted Boltzmann Machine (RBM) to remove KDDcup99 noises and introduce a new data set. Gouveia and Correia [11] used RBM for network IDS to test its capability to learn the complex data. They also proposed a systematic way of dataset learning.

To the best of our knowledge, a comprehensive comparison/evaluation of IDS for WSN-based critical monitoring infrastructures that works for both known and unknown attacks using adaptive machine learning and RBM-based deep learning remains an open issue.

III. A Deep Learning Model for WSN IDS

A. Restricted Boltzmann Machine (RBM) Procedure

The RBM is a neural, energetic network with two layers: visible (V) and hidden (H). The learning procedure is managed by an unsupervised fashion [10]. The RBM permits connections between neurons of the same layer, making it restricted. In RBM, W represents the weights between visible and hidden layers and W_{xy} represents the weight of both visible V_x and hidden H_y units. The energy function of the RBM is shown in Equation (1) below.

$$E(V, H|\Theta) = -\sum_{x=1}^{X} a_x V_x - \sum_{y=1}^{Y} b_y H_y - \sum_{x=1}^{X}\sum_{y=1}^{Y} V_x H_y W_{xy} \quad (1)$$

Θ refers to W_{xy}, a_x, b_y (RBM parameters), a_x and b_y are the visible and hidden biases, and *X* and *Y* are the number of visible and hidden nodes.

The probability of (V, H) formation is calculated as in equation (2) [10].

$$P(V,H) = e^{-E(V,H)} / \sum_{X,Y} e^{-E(V,H)} \tag{2}$$

where $\sum_{X,Y} e^{-E(V,H)}$ refers to the normalization factor that represents all possible configurations, including the visible and hidden elements. With the energy function, the network allocates a probability score to each case in the hidden and visible elements. The probability allocated to a visible element *V* is presented in eq. (3) [10].

$$P(V) = \sum_{Y} P(V,H) = \frac{\sum_Y e^{-E(V,H)}}{\sum_X \sum_Y e^{-E(V,H)}} \tag{3}$$

Likewise, the probability allocated to any hidden element *H* is presented in eq. (4) below [10].

$$P(H) = \sum_{X} P(V,H) = \frac{\sum_X e^{-E(V,H)}}{\sum_X \sum_Y e^{-E(V,H)}}. \tag{4}$$

B. Deep Learning-Based IDS: RBC-IDS

The RBC-IDS consists of the *N* clusters with *C* sensor nodes in each cluster. In each cluster, the Cluster Head (CH) is in charge of sending the sensor directed data to the IDS, which is installed in a central server. The aggregated data then undergoes deep learning-based Restricted Boltzmann Machine IDS, namely the RBC-IDS.

Using the RBC-IDS, as with the ASCH-IDS [2], the CH selection method is accomplished by using the weighted cluster head election technique, which calculates the weight of each sensor node and compares it with the weights of other nodes' [12]. With this method, each sensor is given a weight W_n that is a function of the node (received signal strength (RSS), mobility, and degree). After computing the weight, the node shares it with its ID number, then compares it with the weights of adjacent nodes, such as the sensor node. The *CH* will be the node that achieves the lowest W_n [13]. The election procedure goes through the steps shown in Algorithm 1.

In the RBC-IDS, each CH totals the sensed data from the other sensors in its consistent cluster, and sends this to the server by adopting the data aggregation procedure in [14]. The procedure computes the aggregator trust score based on other sensor trust scores and the trust evaluation between the sensors and the aggregator [14].

In eq. (5), T_{CH} is the trust score of the CH (aggregator), T_n refers to the node *n* trust score and T^n_{CH} is the trust evaluation of the CH and the sensor node *n*.

$$T_{CH} = (\sum_{n=0}^{n-1} (T_n + 1) \cdot T^n_{CH}) / \sum_{n=0}^{n-1} (T_n + 1) \tag{5}$$

In RBC-IDS, the RBM method consists of input layers that contain *x* visible nodes, such as (V_1, $V_2, \ldots, V_x$), hidden layers and the outputs.

Algorithm 1 Weighted CH Selection Pseudo-Code

procedure CH SELECTION TECHNIQUE
Inputs: d_n, δ, $SRSS_n$, M_n, τ_n.
Outputs: W_n.
  for each node *n* **do**
    $d_n \leftarrow$ neighboring sensors' number of *n*
    $\delta \leftarrow$ Capacity of a CH *(number of nodes)*
    $\Delta n \leftarrow$ Degree difference for *n*
    $\Delta n = |d_n - \delta|$
    $SRSS_n \leftarrow$ Sum of n's received signal strength
    $|1/SRSS_n| \leftarrow$ normalized sum of RSS for *n*
    $M_n \leftarrow$ Mobility factor for node *n*
    $\tau_n \leftarrow$ Cumulative time *n*
    $W_n \leftarrow$ Combined weight for node *n*
    $W_n = f_1 \Delta n + \frac{f_2}{|1/SRSS_n|} + f_3 M_n + f_4 \tau_n$
  end for
  Return W_n
  Select the node with a minimum W_n as CH
  Eliminate the elected CH from the nodes set
  Repeat steps for all remaining nodes
  End
end procedure

TABLE I
KDD'99 DATASET ATTACK RECORDS

KDD dataset	DoS	R2L	U2R	Probe
10% KDD	391458	1126	52	4107
Corrected KDD	229853	16347	70	4166

TABLE II
ATTACKS EXAMPLES IN KDD 99 DATASET

Attacks	Examples	Number of attacks
DoS	Smurf, teardrop	6
R2L	passwd,multihop	8
U2R	buffer_overflow	4
Probe	portsweep	4

IV. PERFORMANCE EVALUATION

We evaluated the performance of the deep learning-based IDS (RBC-IDS) and compared it with the previously presented adaptive machine learning-based IDS (ASCH-IDS) [2] with regards to Accuracy Rate (AR%), False Negative Rate (FNR%) and Detection Rate (DR%). In our simulation, we used the Network Simulator-3 (NS-3).

In the RBM, we train a single layer then go through the entire deep network. In our simulation we partition the KDD dataset for training and use the attack records for testing as shown in Table I. Under the simulation settings in Section IV-A, we evaluate the two mechanisms by using KDD'99 as a real attack dataset. Tables I and II present the dataset profile in terms of the number of attack records and attack examples.

We present the performance results in terms of AR%, DR%, FNR, ROC curve and F_1 score characteristics in Section IV-B.

TABLE III
TESTING SETTINGS

Simulation Inputs	Input Value
Visible nodes (x)	41
Hidden layers	3
Sensors number	20
Routing protocol	Hierarchical-DSR (H-DSR)
Clusters number	4
Simulation time	600s
Packet size	250 bytes
Operational area	100m x 100m
Communication range	100m
Dataset	KDD'99
Attack Types	DoS,R2L,U2R,Probe

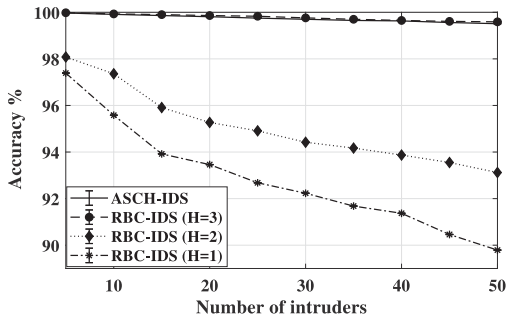

Fig. 1. ARs% comparison of RBC-IDS and ASCH-IDS.

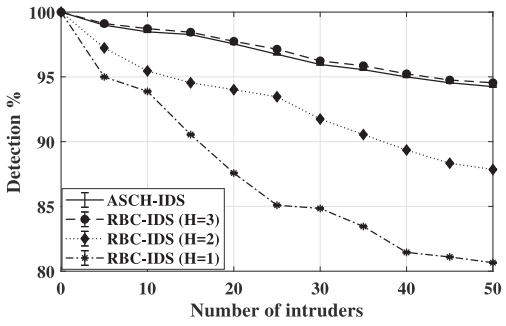

Fig. 2. DRs% comparison of RBC-IDS and ASCH-IDS.

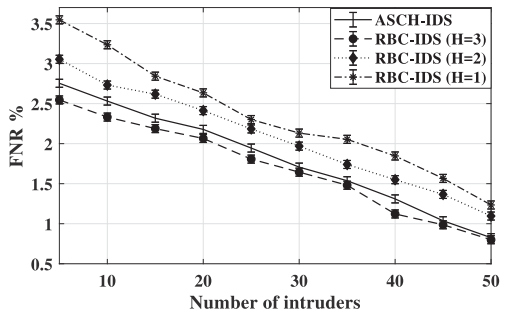

Fig. 3. FN% comparison of RBC-IDS and ASCH-IDS.

A. Simulation Inputs

We simulated a network of twenty sensors that deployed in a WSN and communicated using the Dynamic Source Routing protocol used for Hierarchical representation networks (H-DSR). The tested sensors are deployed in four clusters in an area of 100m x 100m. The figures reflect the median of 10 executions for each scenario, with 95% confidence scale. Table III is a detailed list of the simulation inputs. The RBM model integrated with the simulations consists of one input layer (V_1) with 41 features, 3 hidden layers (H_1, H_2 and H_3) and the output which is classified as normal or malicious.

B. Simulation Results

The presented ASCH-IDS and RBC-IDS are evaluated based on the following criteria: *i*. True Positive (TP) denotes anomalous cases that were correctly classified anomalous, *ii*. False Positive (FP) stands for normal cases that were incorrectly classified anomalous, *iii*. True Negative (TN) denotes normal cases that were classified correctly, and *iv*. False Negative (FN) stands for anomalous cases that were incorrectly classified normal.

1) Accuracy Rate (AR%): The AR% is the ratio of classified incidences that return to True Positive (TP) and True Negative (TN) incidences [15].

AR is presented for different scenarios to trace RBC-IDS performance with different numbers of hidden layers (H) and compare them with ASCH-IDS AR%, as shown in Fig. 1. As the figure shows, the proposed RBC-IDS with $H = 3$ results had the highest *AR* of 99.91%, followed by ASCH-IDS with 99.83%. RBC-IDS with $H = 1$ achieved the least *AR*.

2) Detection Rate (DR%): The DR% represents the behaviours that are accurately recognized as intrusive, and signifies the (*TP*) ratio as displayed in Eq. (6), where *TP* and *FP* denotes the True Positive and False Positive respectively [15]. The *DR%* for RBC-IDS with different hidden layer numbers ($H = 3$, $H = 2$ and $H = 1$) is compared to ASCH-IDS as shown in Fig. (2).

$$DR\% = TP/(TP + FP) \tag{6}$$

Fig. 2 illustrates the *DRs* for the proposed RBC-IDS with different numbers of hidden layers (H), and compares them with ASCH-IDS DR%. The proposed RBC-IDS with H=3 achieves the highest *DR*, followed by ASCH-IDS as shown in Fig. 2.

3) False Negative Rate (FNR%): FNR% returns to the ratio of undesirable sensor behaviours that have been inaccurately classified as non-intrusive [13]. *FN* represents network failure to detect intrusive behaviors, such as the negative activities originated by sensors that are not detected. The FNR% of RBC-IDS and ASCH-IDS are shown in Fig. 3.

In deep learning-based RBC-IDS with $H = 3$, the overall FNR% is mitigated when compared to the case under the ASCH-IDS. RBC-IDS with $H = 1$ achieves the highest FNR%, which represents the lowest performance.

4) Receiver Operating Characteristic Curve: The ROC curve displays the ratio between sensitivity (*TP*) and the *FP* ($1-$*Specificity*). Sensitivity-specificity is represented by the area under the curve, with the larger area reflecting the best performance. ROC curves were plotted for RBC-IDS with different numbers of hidden layers such as ($H = 1$, $H = 2$ and $H = 3$), compared to ASCH-IDS as shown in Fig. 4.

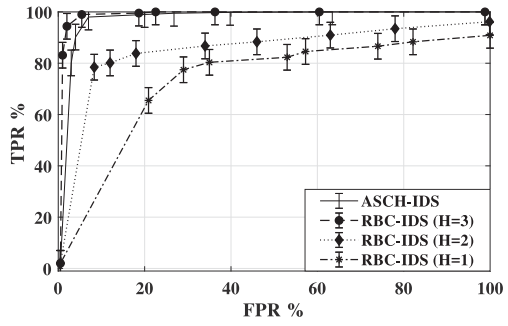

Fig. 4. ROC comparison of RBC-IDS and ASCH-IDS.

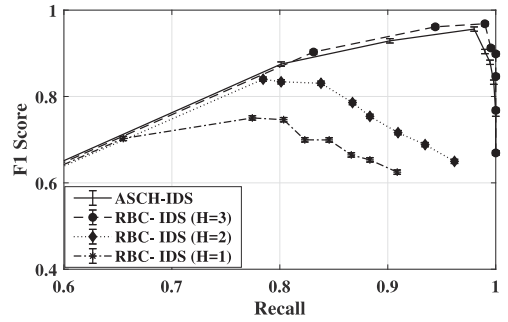

Fig. 5. F_1 Score comparison of RBC-IDS and ASCH-IDS.

TABLE IV
TRAINING TIME AND TESTING TIME COMPARISON

Tested procedure	Training time (s)	Testing time (s)
RBC-IDS	31.5	1.62
ASCH-IDS	17.1	0.86

With ROC curves, the overall RBC-IDS performance can be enhanced when $H = 3$.

5) F_1 Score Curve: The F_1 score measures test accuracy [16], studying the precision-recall of the test in order to calculate its F score. The precision is the number of true positive incidences divided by all positive incidences, which is formulated as *TP*/(*TP*+*FP*). The recall is formulated as *TP*/(*TP*+*FN*), which represents the number of true positive incidences divided by all *actually* positive instances. The F_1 score is the harmonic average of precision and recall [16].

Table IV shows the detection time (training and testing) times of RBC-IDS and ASCH-IDS procedures. It is clear that the detection time (training and testing) for the RBC-IDS procedure is higher than that of the ASCH-IDS procedure.

V. CONCLUSION

In this letter, we presented a feasibility analysis of a deep learning-based IDS known as the Clustered Restricted Boltzmann Machine-Intrusion Detection System (RBC-IDS), and compared to an adaptive machine learning-based IDS approach [2]. We also compared the RBC-IDS performance with different numbers of hidden layers against the ASCH-IDS through simulations, and verified that the proposed RBC-IDS has a $\approx 99.12\%$ detection rate and $\approx 99.91\%$ accuracy rate with three hidden layers ($H = 3$), with intrusive behaviours present in the tested WSN. We have shown that the adaptive machine learning-based solution performs at the same rate as the deep learning-based solution, whereas adopting a machine learning-based IDS framework leads to approximately half the detection time of the deep learning-based RBC-IDS framework. Our future agenda includes, extending the presented IDS to larger networks with more sensors.

REFERENCES

[1] N. Chand, P. Mishra, C. R. Krishna, E. S. Pilli, and M. C. Govil, "A comparative analysis of SVM and its stacking with other classification algorithm for intrusion detection," in *Proc. Int. Conf. Adv. Comput. Commun. Autom. (ICACCA) (Spring)*, Apr. 2016, pp. 1–6.

[2] S. Otoum, B. Kantarci, and H. T. Mouftah, "Adaptively supervised and intrusion-aware data aggregation for wireless sensor clusters in critical infrastructures," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.

[3] K. Alrawashdeh and C. Purdy, "Toward an online anomaly intrusion detection system based on deep learning," in *Proc. IEEE Int. Conf. Mach. Learn. Appl.*, Dec. 2016, pp. 195–200.

[4] N. D. Lane and P. Georgiev, "Can deep learning revolutionize mobile sensing?" in *Proc. Int. Workshop Mobile Comput. Syst. Appl.*, 2015, pp. 117–122.

[5] M. E. Haque and T. M. Alkharobi, "Adaptive hybrid model for network intrusion detection and comparison among machine learning algorithms," *Int. J. Mach. Learn. Comput.*, vol. 5, no. 1, pp. 17–23, Feb. 2015.

[6] M. Z. Alom, V. Bontupalli, and T. M. Taha, "Intrusion detection using deep belief networks," in *Proc. Nat. Aerosp. Electron. Conf. (NAECON)*, Jun. 2015, pp. 339–344.

[7] M. A. Salama, H. F. Eid, R. A. Ramadan, A. Darwish, and A. E. Hassanien, "Hybrid intelligent intrusion detection scheme," in *Soft Computing in Industrial Applications*, vol. 96. Berlin, Germany: Springer, 2011, pp. 293–303.

[8] U. Fiore, F. Palmieri, A. Castiglione, and A. De Santis, "Network anomaly detection with the restricted Boltzmann machine," *Neurocomputing*, vol. 122, pp. 13–23, Dec. 2013.

[9] Y. Li, R. Ma, and R. Jiao, "A hybrid malicious code detection method based on deep learning," *Int. J. Security Appl.*, vol. 9, no. 5, pp. 205–216, 2015.

[10] S. Seo, S. Park, and J. Kim, "Improvement of network intrusion detection accuracy by using restricted Boltzmann machine," in *Proc. 8th Int. Conf. Comput. Intell. Commun. Netw. (CICN)*, Dec. 2016, pp. 413–417.

[11] A. Gouveia and M. Correia, "A systematic approach for the application of restricted Boltzmann machines in network intrusion detection," in *Advances in Computational Intelligence*. Cham, Switzerland: Springer, 2017, pp. 432–446.

[12] F. Belabed and R. Bouallegue, "An optimized weight-based clustering algorithm in wireless sensor networks," in *Proc. Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, 2016, pp. 757–762.

[13] S. Otoum, B. Kantarci, and H. T. Mouftah, "Detection of known and unknown intrusive sensor behavior in critical applications," *IEEE Sensors Lett.*, vol. 1, no. 5, pp. 1–4, Oct. 2017.

[14] W. Zhang, S. K. Das, and Y. Liu, "A trust based framework for secure data aggregation in wireless sensor networks," in *Proc. IEEE ComSoc Sensor Ad Hoc Commun. Netw.*, 2006, pp. 60–69.

[15] B. M. Beigh and M. A. Peer, "Performance evaluation of different intrusion detection system: An empirical approach," in *Proc. Int. Conf. Comput. Commun. Informat.*, Jan. 2014, pp. 1–7.

[16] M. Elhamahmy, N. Hesham, and A. Imane, "A new approach for evaluating intrusion detection system," in *Artif. Intell. Syst. Mach. Learn.*, vol. 2, no. 11, pp. 290–298, Dec. 2010.