# Deep Feature Extraction for multi-Class Intrusion Detection in Industrial Control Systems

Sasanka Potluri and Christian Diedrich

***Abstract*—In recent days, network based communication is more vulnerable to outsider and insider attacks due to its wide spread applications in different domains. Intrusion detection is a key task for defense-in depth strategy of the communication networks. In order to defend properly against growing threats, Industrial Control Systems (ICS) and Supervisory Control and Data Acquisition (SCADA) need to incorporate this technology. Intrusion Detection System (IDS), a software application or a hardware which is able to monitor the network traffic and find abnormal activities in the network. Due to raise in the network bandwidth and network data a deep packet inspection is necessary to extract proper features and identify the attacks. Deep Neural Networks (DNN) a deep learning approach is used in this paper to identify the different types of attacks in the network packets. NSL-KDD dataset is used to evaluate the effectiveness of the proposed IDS. Experimental results show that the features extracted using DNN provides a better classification accuracy than the conventional machine learning techniques.**

***Index Terms*—Deep learning, intrusion detection system, industrial control system, network security, deep neural networks.**

## I. Introduction

With the rapid application of the network based communication in industries, the security related problems appear to be more serious. Attaching Industrial Control Systems (ICS) or Supervisory Control and Data Acquisition (SCADA) networks to the Internet benefited companies and engineers a lot. As a consequence, number of threats like Cyberattacks and Malware attacks were faced by ICS and SCADA networks. Some common attack types on communication network are vulnerabilities, SYN flooding, Distributed Denial-of-Service (DDOS), surfing and the list goes on. As ICS/SCADA systems mainly control critical infrastructure, failure in such systems may endanger people's health and safety, damage industrial facilities and produce financial loss. One of the approaches to securing the communication network attached ICS is a Network based Intrusion Detection System (NIDS). Ref. [1] summarizes the challenges and different scientific efforts to improve the security in ICS.

Intrusion detection refers to the detection of malicious activity (attacks, break-ins, penetrations and other forms of computer abuse) in a computer related systems or in the communication networks. An intrusion can be sometimes identified as a completely different behavior from the normal or sometimes hard to distinguish it from normal behavior. An Intrusion Detection System (IDS) protect from attacks and penetration attempts on the network. IDS is an individual or a combination of several individual intrusion detection techniques together such as signature analysis, traffic monitoring, anomaly detection etc. Signature analysis or traffic monitoring searches for well-known patterns of attacks in the network. Therefore, it can only detect an attack if there is an accurate matching behavior against the stored or known patterns termed as signatures. In the ICS context, the number of known attacks is relatively small, thus there are still few attack signatures available. This implies that the effectiveness of these approaches has less effect in ICS environment in relation to IT systems. On the other hand, anomaly detection establishes a normal activity profile for a system. This technique evolves itself by understanding and gathering the information about the system and determines the behavior of the system. However, a large number of false positives constraints the wide deployment of anomaly detection systems in real-world environment.

An important challenge to develop an efficient and flexible NIDS lies in the identification of proper features in network packets. Feature extraction and learning from the network traffic dataset for anomaly detection is difficult. Two main reasons for the complexity of feature extraction are: One, the features extracted for one class of attacks may not be suitable for other categories of attacks. Second, continuously evolving and changing attack scenarios and strategies. Due to these reasons, a complex and meaningful feature extraction is necessary to identify the multiple attack classes in the ICS networks.

Various Machine Learning (ML) techniques have been used to develop NIDS such as Artificial Neural Networks (ANN's), Support Vector Machines (SVM's), Naive-Bayesian (NB) and Self Organizing Maps (SOM's). All these ML techniques train in a supervised or unsupervised manner to identify the normal and attack packets in network traffic. With the raise of network bandwidth and traffic speed the problems with traditional IDS are packet loss, slow detection and higher response time to deal with the huge network data. Recent advances in deep learning methods based on ANN's grabs our interest for the development of IDS. These techniques have already led to breakthroughs in longstanding artificial intelligence tasks such as speech, image and text recognition, language translation etc. Ref. [2] already envisioned that deep learning based approaches can help to overcome the challenges to develop an efficient IDS.

In this paper, NSL-KDD dataset is used to train and test

Manuscript received February 5, 2017; revised April 15, 2017. This work is being supported by the project is Secure funded by Federal Ministry for Economic Affairs and Energy (BMWi) and Federation of Industrial Research Associations "Otto von Guericke" (AIF), a collaborative project between Otto-von-Guericke University Magdeburg (Institute for Automation Engineering) and ifak systems GmbH, Magdeburg.

Sasanka Potluri and Christian Diedrich are with Institute for Automation Engineering, Otto-von-Guericke University Magdeburg, 39106 Magdeburg, Germany (e-mail: sasanka.potluri@ovgu.de, christian.diedrich@ovgu.de).

DOI: 10.7763/IJCTE.2017.V9.1169

our approach [3]. NSL-KDD is a benchmark dataset for various IDS evaluations. Stacked-Auto-encoders, a deep learning technique is used to extract the complex relations out of the NSL-KDD dataset. Later we trained the soft-max regression layer to form a Deep Neural Network (DNN) for IDS. DNN exhibit major differences from traditional approaches for classification. Primarily, they are deep in architecture with multiple hidden layers which makes them to learn more complex models than shallow once [4]. Next, their expressivity and robust training algorithms allows powerful learning and complex object representations.

The DNN based IDS classifies the different attack classes represented in NSL-KDD dataset. The detection accuracies of identifying the individual attacks is evaluated. The detection accuracy of DNN depends on the feature selection based on the network size, the number of hidden layers and their training epochs. In addition, to the handling of the huge NSL-KDD dataset, there is no need to manually design a model which extracts their complex relations explicitly. This simplicity has the advantage of easy applicability to a wide range of attack classes and also shows better detection performance across a wider range of datatypes. By configuring and tuning the DNN parameters, the developed DNN based IDS classifies all the attack classes present in NSL-KDD dataset and improve the detection accuracies in comparison to conventional methods.

## II. Related Work

The concept of using intrusion detection in information security came into mainstream with DARPA Intrusion Detection Evaluation [5] released in 1998 and 1999 in conjunction with the MIT. Several researchers used different existing dataset like KDD, NSL-KDD etc. to evaluate the performance of their developed IDS. A detailed analysis on different datasets for intrusion detection is mentioned in [6]. The drawbacks of the existing KDD cup 99 dataset discussed by several researchers [7] lead to the development of NSL-KDD dataset. It contains essential record of the complete KDD dataset.

In this paper, we discuss the latest machine learning approaches used for IDS development using only the NSL-KDD dataset. Therefore, a dataset referred from this point should be considered as NSL-KDD dataset only. Many researchers carried out different analysis on NSL-KDD dataset and employed different techniques and tools with a common aim to develop an effective IDS. A detailed analysis on NSL-KDD dataset using various machine learning techniques with Waikato Environment for Knowledge Analysis (WEKA) tool is discussed in [8].

The major motivation behind using the machine learning techniques for anomaly based intrusion detection is to identify the complex relations out of the input data and extract the necessary feature to identify the normal and attack packets in the network traffic.

While NSL-KDD data set is huge and has several features (41), some of the researchers identified different ways for feature selection to efficiently detect the attacks in NSL-KDD data. Ref. [9] proposed an IDS framework by selecting 23 features out of 41 features using Information Gain (IG) techniques and clustered the dataset into either normal or attack type using the K-means clustering. The detection accuracies were improved when compared to using all 41 features by k-means clustering. The identification of different attack classes in the network packets is not considered in this work. In [10] six attributes corresponding to very low information gain are removed in NSL-KDD dataset. They also eliminated 3751 objects from the testing dataset as their attack classes are not present in train dataset to increase the detection accuracy. The training and testing of the proposed ML approach was performed using 10% of the train dataset only. Neural Networks with Indicator Variable using Rough Set (NN-IVRS) for attribute reduction technique was proposed in [11]. They have achieved better detection accuracies but classification of different attack classes is not considered. Due to this reason, they have only normal and attack classes which improves the detection accuracy due to generalization. Enhanced Resilient Backpropagation (ERBP) was proposed for IDS by [12]. They classified all 5 classes and achieved better detection accuracies for the DoS, U2R and R2L attacks. But the detection accuracy of the normal data packets is low due to consideration of same size of data for normal as well as for attack. The classification of normal data is more important when considering huge amount of input data otherwise it raises more frequent alarms and halt the entire system. Ref. [13] proposed an unsupervised neural network Adaptive Resonance Theory (ART) and Self-Organizing Maps (SOM) for IDS. Their approaches have achieved a good detection accuracy of the normal data but the attack classification rate was quite low. Deep Belief Network (DBN) for IDS was proposed by [14] and explained the efficiency of achieving higher accuracies. They performed the training operation with 20%, 30% and 40% of the NSL-KDD train dataset and tested it with the same. Finally, the most recent approach of implementing deep learning for NIDS proposed a Self-Taught Learning (STL) and Soft-max regression (SMR) approach [2]. Their Precision, Recall and F-Measure values were higher for 2 class attack classification but due to use of single spare auto-encoder, the extracted features are not sufficient to classify the all 5-classes in NSL-KDD data with high precision.

There exist some drawbacks from many existing approaches. Firstly, many researchers used the training data for both training and testing purpose the efficiency of developed IDS. This approach does not provide an accurate classification result as the amount of data for different attack classes is different from training to testing dataset. Secondly, researchers considered only selected part of the dataset to train and test. This improves the detection accuracies but the overall efficiency by using the complete dataset may decrease by using the proposed approaches. Finally, most of the literature concentrated on classifying either as an attack or as normal network packet. Due to this generalization, the attack identification accuracy is higher but in real time scenarios, the preventive measures after attack identification depends on the type of identified attack.

To overcome the specified draw backs of different feature selection methods for multi-class attack classification on NSL-KDD dataset we developed the DNN based IDS. Another downside from most of the researchers is, they are taking the confusion matrix of the attack identification and provide an overall detection accuracy. This may give a false impression of identifying the different attack classes with the same classification accuracy. As the NSL-KDD dataset

consists of different attack classes with different data samples, it is hard for a machine learning technique to learn all the attack classes efficiently. Due to this reason the detection accuracies of different attack classes need to be mentioned separately. The generated model is initially trained and tested with all 41 features of the dataset to detect the different attack classes. Later in order to verify the effectiveness, we trained and tested the DNN with selected feature sets used by different researchers. This evaluates the capabilities of DNN based IDS in identifying appropriate relations between the input features for attack classification. Several researches also discussed about the importance and challenges on implementing the IDS in ICS. The deep feature extraction and IDS proposed in this paper is applicable to analyze the network packets in ICS without slowing the network performance.

## III. Proposed Intrusion Detection Architecture

### A. NSL-KDD Dataset

NSL-KDD dataset have 41 attributes unfolding different features of the traffic flow and a label is assigned to each either as a particular attack type or as a normal data. The details of the attributes namely the attribute name, their description and sample data is given in [15]. The features in the NSL-KDD dataset are of different datatypes.

TABLE I: Features with Different Datatypes in NSL-KDD Dataset

| Datatype | Features |
|---|---|
| Nominal | 2, 3, 4 |
| Binary | 7, 12, 14, 15, 21, 22 |
| Numeric | 1,5,6,8,9,10,11,13,16,17,18,19,20,23, 24,25,26,27,28,29,30,31,32,33,34,35, 36,37,38,39,40,41 |

TABLE II: Attack Types into Different Attack Classes

| Attack Class | Attack Types |
|---|---|
| DoS | Back, Land Neptune, Pod, Smurf, Teardrop, Apache 2, Udpstorm, Processtable, Worm (10) |
| Probe | Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint (6) |
| R2L | Guess_Password, Ftp_write, Imap, Phf, Multihop, Waremaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendfmail, Named (16) |
| U2R | Buffer_overflow, Loadmodule, Rotkit, Perl, Sqlattack, Xterm, Ps (7) |

The datatypes and the corresponding feature numbers are given in the Table I. Apart from normal data, records for 39 different attack types exist in NSL-KDD dataset. All these attack types were grouped into four attack classes. The summary of attack classes and attack types is given in Table II and Table III gives an overview on the number and percentage of records present in train and test data of the NSL-KDD dataset. Both the train and test data set in NSL-KDD dataset have labels mentioning either attack or normal data. These are helpful in performing supervised training and to evaluate the detection accuracy.

TABLE III: Overview on NSL-KDD Dataset

| Data set type | Numbers of data samples | | | | | |
|---|---|---|---|---|---|---|
| | Records | Normal | DoS | Probe | U2R | R2L |
| NSL-KDD Train | 125973 | 67343 | 45927 | 11656 | 52 | 995 |
| | % | 53.46 | 36.45 | 9.25 | 0.04 | 0.79 |
| NSL-KDD Test | 22543 | 9711 | 7458 | 2421 | 200 | 2754 |
| | % | 43.08 | 33.08 | 10.74 | 0.89 | 12.22 |

### B. Deep Neural Network Architecture for IDS

Various steps involved in extracting the necessary features out of NSL-KDD dataset and later training and testing the DNN for classifying normal and different attack classes in the dataset was shown in Fig. 1.

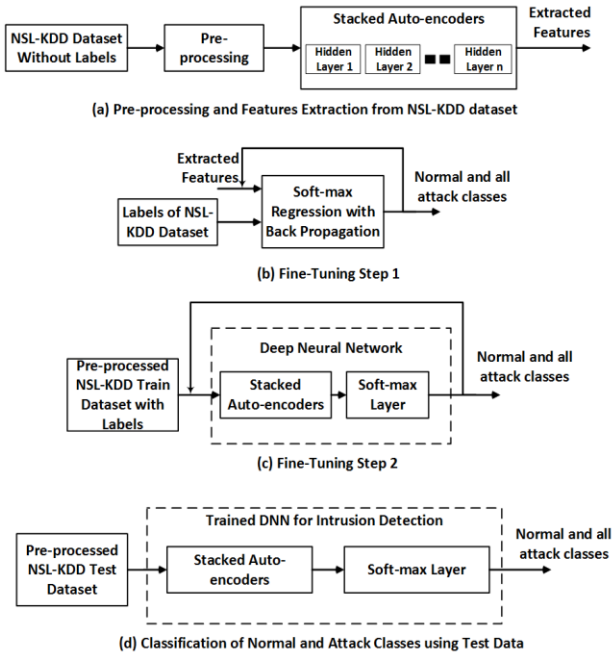


Fig. 1. Various steps involved in DNN based IDS.

#### a) Pre-processing and feature extraction

The neural network based classification uses only numerical values for training and testing. Hence a pre-processing stage is necessary to convert the non-numerical attributes to numerical values. Two main tasks of pre-processing are:

1) Converting the non-numerical attributes in the dataset to numerical values. The features 2, 3 and 4 represent the protocol type, service and flag. These attributes in the NSL-KDD train and test data set were converted to numerical values (e.g. Protocol type - TCP = 1, UDP = 2 and ICMP = 3).
2) Converting the attack name at the end of the dataset into its numeric categories. 1,2,3,4 and 5 were assigned to normal, DoS, Probe, R2L and U2R respectively.

Since the features of the NSL-KDD dataset have either discrete or continuous values, the ranges of the features value were different and this made them incomparable. Therefore, the features were normalized by using min-max normalization to map all the different values for individual attributes to range between [0 ,1].

Feature extraction process by the auto-encoders is done without using the labels. An auto-encoder will attempt to replicate its input at its output. Thus the size of its output will

be the same as the size of its input. When the number of neurons in the hidden layer is less than the size of the input the auto-encoder learns a compressed representation of the inputs.

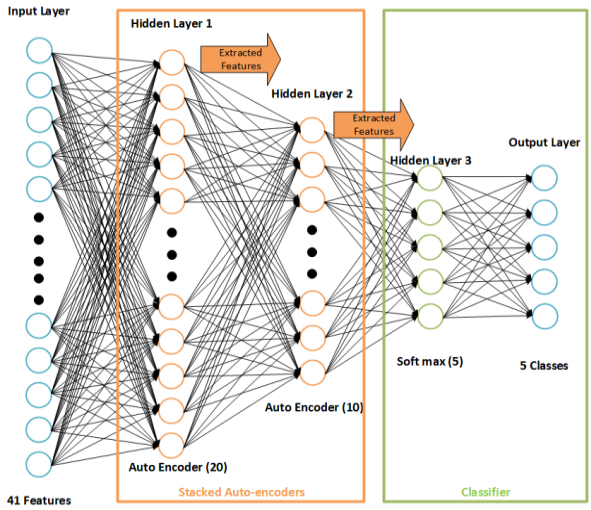


Fig. 2. DNN architecture in detail.

In our case as shown in Fig. 2, the 41 feature from the NSL-KDD dataset were converged to 20 feature set. The 20-dimensional output from the *Hidden Layer 1* of the auto-encoder is a compressed version of the input. The next auto-encoder is trained with the set of these generated features extracted from the training data. In-order to extract the features out the trained dataset, first we train the auto-encoders in an unsupervised manner and later the features are extracted from the trained auto-encoder. The feature extraction process continues to the next layer. After training the first auto-encoder, the training of second auto-encoder is done. As shown in Fig. 2, the extracted features from *Hidden Layer 1* are used as an input to the *Hidden Layer 2*. At this layer the 20 features were further compressed to 10 features. The hidden layer 1 and 2 form a stacked auto-encoder structure and extracts the complex features out of the NSL-KDD dataset. These features were then given to soft-max layer for classification purpose.

*b) Fine tuning step 1*

After extracting the necessary features, the soft-max layer is trained to classify the attack and normal network packets. Unlike the auto-encoder's, supervised learning is used for the soft-max layer using labels of the training data. The process of fine tuning step 1 is represented in Fig. 1 (b). The inputs to the soft-max layer are the features extracted from the stacked auto-encoder's.

*c) Fine tuning step 2*

After training the soft-max layer the complete DNN is further fine-tuned by performing backpropagation on all multiple layers. As shown in Fig. 1 (c), the supervised backpropagation refines the features in the intermediate layers so that they become more relevant to the intrusion detection task.

*d) Classification and testing*

After the training of complete DNN for intrusion detection, it is capable of detecting the attacks in the NSL-KDD dataset. The testing of the trained DNN is done with additional NSL-KDD test dataset and the attack detection accuracies were measured. This is shown in Fig. 1 (d).

## IV. Evaluation and Results

### A. IDS Implementation

The trained DNN is capable of identifying different attack classes in NSL-KDD dataset. We compared the efficiency of our feature extraction techniques with the different feature sets discussed in the literature. As NSL-KDD consists of 41 features we trained initially with all 41 feature as shown in Fig. 2 and analyzed the results. The number of auto-encoders in the stack for feature extraction varies with the number of features considered for training the DNN. The training epochs of the individual layers were carefully tuned for best feature extraction to avoid over fitting and under fitting of the training process. Apart from the entire 41 features in the NSL-KDD dataset, we also considered the reduced feature set discussed in the literature to analyze the capabilities of the DNN for feature extraction and attack identification. Table IV gives the information about the number of hidden layers required to build the IDS using DNN for different selected feature set. Apart from all 41 features, 35 [10] and 23 [9] features also require 3 hidden layers which include 2 auto-encoders and 1 soft-max layer for classification of attacks and normal data.

### B. Accuracy Metrics

We evaluated the performance of the DNN based IDS on the following attributes resulted from training and testing dataset of NSL-KDD. These values are True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN). These entries were used to compute the following different metrics

*Accuracy:* Defined as the percentage of correctly classified records over the total number of records.

*Precision (P):* Referred as Positive Predictive Value (PPV) defined as the % ratio of the number of TP records divided by the sum of TP and FP classified records.

$$P = \frac{TP}{TP+FP} * 100\% \quad (1)$$

*Recall (R):* Referred as the TP rate as sensitivity defined as the % ratio of number of TP records divided by the sum of TP and FN classified records

$$R = \frac{TP}{TP+FN} * 100\% \quad (2)$$

*F-Measure (F):* A measure to represent test accuracy defined as the harmonic mean of precision and recall and represents a balance between them.

$$F = \frac{2*P*R}{P+R} \quad (3)$$

### C. Performance Evaluation

We implemented the DNN based IDS for multi-class attack classification. It classified the input data whether it belongs to the normal or the 4 attack classes. We evaluated the classification accuracy for all the 41 features and other

features sets used in the literature. The accuracy of detection along of individual classes along w.r.t the feature set was shown in Fig. 3. The detection accuracy of different attack classes was higher using DNN when compared to other machine learning techniques [16].

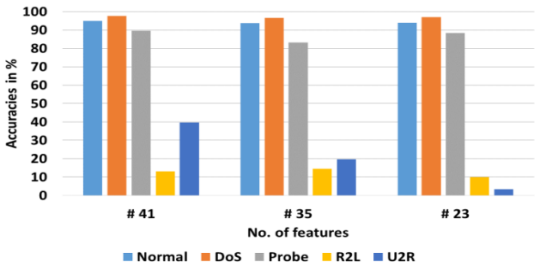


Fig. 3. Multi-class detection accuracy with different features.

The SI unit for magnetic field strength $H$ is A/m. However, if you wish to use units of T, either refer to magnetic flux density $B$ or magnetic field strength symbolized as $\mu_0 H$. Use the center dot to separate compound units, e.g., "A·m$^2$."

The detection accuracies of the Normal and DoS attack classes are high with different feature sets while there exist several samples mainly related to these two types. Detection of probe attack is higher with 41 feature while with some eliminated feature set the detection accuracy also decreases. The accuracy of detecting R2L and U2R is still lower because of very limited data sample available in NSL-KDD dataset in relation to other attacks.

As mentioned earlier, the accuracy metric needs to be evaluated for individual attack classes separately. The accuracy metrics for individual classification is shown in the following Table IV.

In order to evaluate the efficiency of our approach with other existing approaches, the following Fig. 4 provides the overall detection accuracies of the existing techniques and our approach.

TABLE IV: ACCURACY METRICS FOR MULTIPLE CLASSES

| | Normal | DoS | Probe | R2L | U2R |
|---|---|---|---|---|---|
| Precision | 93.4 | 96.2 | 87.84 | 39.91 | 40.38 |
| Recall | 93.2 | 97.6 | 86.34 | 12.98 | 39.62 |
| F-Measure | 93.2 | 96.89 | 87.08 | 19.58 | 39.99 |

The red line in the Fig. 4 gives a comparison of the detection accuracy of our approach with the existing approaches. There exist some techniques which provide better detection accuracies than our approach which crosses the red line. This is due to the use of single attack class for classification or using the same training dataset for training and testing of the machine learning technique.

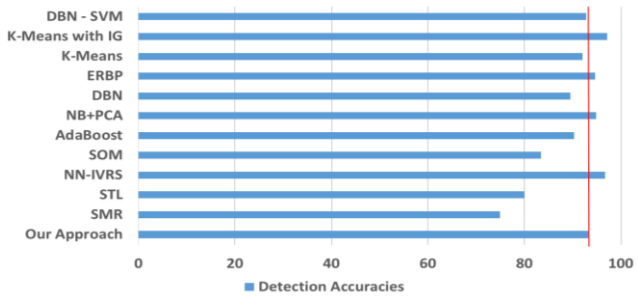


Fig. 4: Benchmarking existing approaches with our approach.

## V. CONCLUSION AND FUTURE WORK

This research mainly focuses on the efficient feature extraction from the input dataset for multi-class attack identification. DNN based architecture is used for this purpose. The packet handling capabilities of the DNN based IDS approach provide a way for its implementation in ICS/SCADA. Experimental results showed that the proposed algorithm gives better and robust feature extraction. It was able to reduce the number of features resulting in the improved detection accuracy of multiple attack classes with all (41) features in the dataset. The obtained results show that the proposed approach is reliable and efficient in intrusion detection for a set of attack classes with required number of samples for training (see DoS and Probe attacks in Figure) and was unable to effectively classify the attack classes (see R2L and U2R in Table III) with low number of samples for training. Despite good detection accuracy for DoS and Probe attacks, due to lack of sufficient amount of data related to R2L and U2R attacks the overall detection accuracy is dropped. A proper dataset with sufficient number of samples needs to be developed for individual attack classes for better training and proper feature extraction. The training of each individual hidden layer will yield in better feature selection process but it consumes a lot time. Parallelization of the DNN training will reduce the computation time in the training process. In our future work, multi-core CPU's and GPU's are considered to effectively decrease the training time of the DNN. Apart from stacked-auto-encoder's, there exists other deep feature extraction techniques such as deep belief networks need to be considered and evaluated. A hybrid machine learning algorithms will also be considered in future for better feature extraction and improved detection accuracies.

## REFERENCES

[1] D. Hadziosmanovic, D. Bolzoni, S. Etalle, and P. Hartel, "Challenges and opportunities in securing industrial control systems," in *Proc. 2012 Complex. Eng. (COMPENG)*, 2012, pp. 1–6.

[2] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in *Proc. 9th EAI Int. Conf. Bio-inspired Inf. Commun. Technol. (formerly BIONETICS)*, 2016.

[3] Information security centre of excellence (ISCX). *UNB ISCX NSL-KDD DataSet.* [Online]. Available: http://www.unb.ca/research/iscx/dataset/iscx-NSL-KDD-dataset.html

[4] C. Szegedy, "Deep neural networks for object detection," Nips 2013, pp. 1–9, 2013.

[5] M. L. Laboratory. DARPA intrusion detection data sets. [Online]. Available: https://www.ll.mit.edu/ideval/data/

[6] S. K. Sahu, S. Sarangi, and S. K. Jena, "A detail analysis on intrusion detection datasets," in *Proc. Souvenir 2014 IEEE Int. Adv. Comput. Conf. IACC*, 2014, pp. 1348–1353.

[7] J. McHugh, "Testing Intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory," *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 4, pp. 262–294, 2000.

[8] D. A. M. S. Revathi, "A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection," *Int. J. Eng. Res. Technol.*, vol. 2, no. 12, pp. 1848–1853, 2013.

[9] D. Mahmood and M. Hussein, "Feature based unsupervised intrusion detection," *Waset. Org*, vol. 8, no. 9, pp. 1515–1519, 2014.

[10] P. Gogoi, B. Borah, and D. K. Bhattacharyya, "Network anomaly identification using supervised classifier," *Inform.*, vol. 37, no. 1, pp. 93–105, 2013.

[11] R. A. Sadek, M. S. Soliman, and H. S. Elsayed, "Effective anomaly intrusion detection system based on neural network with indicator variable and rough set reduction," vol. 10, no. 6, pp. 227–233, 2013.

[12] R. S. Naoum, N. A. Abid, and Z. N. Al-sultani, "An enhanced resilient backpropagation artificial neural network for intrusion detection system," *Int. J. Comput. Sci. Netw. Secur.*, vol. 12, no. 3, pp. 11–16, 2012.

[13] M. Amini, R. Jalili, and H. R. Shahriari, "RT-UNNID: A practical solution to real-time network-based intrusion detection using unsupervised neural networks," *Comput. Secur.*, vol. 25, no. 6, pp. 459–468, 2006.
[14] Z. Alom, V. Bontupalli, and T. M. Taha, "Intrusion detection using deep belief networks," pp. 339–344, 2015.
[15] L. Dhanabal and S. P. Shantharajah, "A study on NSL-KDD dataset for intrusion detection system based on classification algorithms," vol. 4, no. 6, pp. 446–452, 2015.
[16] A. P. O. K. Kruti Choksi and B. Shah, "Improving user-to-root and remote-to-local attacks using growing hierarchical self organizing map," vol. 9655, no. 6, pp. 611–618, 2015.

**Sasanka Potluri** was born in Vijayawada, India. He has bachelors of technology degree in electronics and communication engineering from Jawaharlal Nehru Technology University, Kakinada, India in 2009. Later did master of sciences in information technology in 2011 from Alpen Adria University Klagenfurt, Austria.

He worked as a research assistant in Faculty of Computer Science in Otto-von-Guericke University Magdeburg for 3 years on Autonomous cars safety in an EU Project Karyon. Since 2015 he is working as a research assistant in Institute for Automation Engineering in Otto-von-Guericke University Magdeburg. His main research focus is on machine learning and cyber securtiy in automation.

**Hristian Diedrich** has a German diplom ingenieur degree in electrical engineering with the option automation since 1985 and got his Ph.D in 1994 in the field of semi-formal specification of fieldbus interfaces and fieldbus profiles. His activity field covers the entire engineering life cycle of field devices of production systems. He has worked in many German and European research and development projects (main topics are industrial communication, engineering of automation systems, formal description methods as well as information and knowledge modelling) and is active in national and international standardisation activities.

He is deputy head of ifak e.V. Magdeburg since 2005 and holds the chair of "Integrated Automation" at the Otto-von-Guericke-University of Magdeburg-Germany since April 2006. He is head of the Institute for Automation Engineering at Otto-von-Guericke-University of Magdeburg.