International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Deep Learning Approach for Shrewd Intrusion Detection System

Dr. Putti. Srinivasa Rao, Chittimalla Mithun Kumar, Suryadevara Anusha, Komati Pooja, Kashi Reddy Akshitha

Professor, Department of CSE, JB Institute of Engineering and Technology, Telangana, India. yourpsr@gmail.com
Department of CSE, JB Institute of Engineering and Technology, Telangana, India. mithun.chittimalla@gmail.com
Department of CSE, JB Institute of Engineering and Technology, Telangana, India. anushasuryadevara21@gmail.com
Department of CSE, JB Institute of Engineering and Technology, Telangana, India. poojadec0212@gmail.com
Department of CSE, JB Institute of Engineering and Technology, Telangana, India. akkireddy125@gmail.com

Abstract: In the realm of cybersecurity, the continuous evolution of attack strategies demands robust and adaptive Intrusion Detection Systems (IDS). This paper presents a comprehensive study on enhancing IDS performance through a deep learning framework. Initially, classical algorithms including Support Vector Machines (SVM) and Random Forest were evaluated, revealing limitations in detecting dynamic attacks without prior training. The preprocessing phase involved converting categorical data into numerical values and categorizing attacks into distinct classes, laying the foundation for model training. Moreover, the study extends beyond conventional techniques by introducing Convolutional 2D Neural Networks (CNN2D), leveraging multilayer filtering to select optimal features from the dataset. Results demonstrate significant accuracy improvement, with CNN2D achieving a remarkable 95% accuracy rate. Furthermore, an extension incorporates Extreme Machine Learning (EML) algorithms and parallel processing techniques to further enhance accuracy and reduce execution time. The EML algorithm showcases superior performance in accuracy, while parallel processing demonstrates efficiency gains. Through systematic experimentation and analysis, this study underscores the efficacy of deep learning methodologies in fortifying IDS against evolving cyber threats. The amalgamation of DNN, CNN2D, and EML algorithms, alongside parallel processing, presents a formidable defense mechanism against intrusions, ensuring robust detection capabilities and efficient computational processing. These advancements mark a significant stride towards developing intelligent IDS systems capable of adapting to dynamic cyber landscapes, thereby safeguarding critical network infrastructures with heightened accuracy and efficiency.

Index terms *– Deep Learning, Intrusion Detection System, SVM, CNN2D, EML, DNN.*

1. INTRODUCTION

In the digital age, where cyber threats loom large and incessantly evolve, the integrity of network infrastructures hinges upon the efficacy of Intrusion Detection Systems (IDS). With the escalating sophistication of malicious actors and the ever-expanding attack surface, the imperative to fortify these systems with advanced methodologies has never been more pressing. This paper embarks on a journey to explore the frontier of IDS enhancement, leveraging the prowess of deep learning to bolster

detection accuracy and computational efficiency. Traditional IDS methodologies, epitomized by algorithms like Support Vector Machines (SVM) [11] and Random Forest [21, 32], have long served as stalwart guardians against cyber intrusions. However, their efficacy in dynamically detecting emerging threats without prior training has been called into question. As such, there arises a crucial need for more adaptive and robust detection mechanisms capable of navigating the intricate landscape of contemporary cyber warfare. At the heart of this study lies the adoption of Deep Neural Network (DNN) algorithms, representing a paradigm shift towards more sophisticated and nuanced detection frameworks. By harnessing the power of deep learning, we aim to transcend the limitations of conventional approaches, paving the way for heightened accuracy and adaptability in intrusion detection. Moreover, this exploration extends beyond the realm of traditional deep learning architectures by delving into the realm of Convolutional 2D Neural Networks (CNN2D). Through the application of multilayer filtering techniques, CNN2D promises to extract optimal features from complex datasets, thereby augmenting detection capabilities to unprecedented levels. Furthermore, the integration of Extreme Machine Learning (EML) algorithms and parallel processing techniques serves to amplify the efficacy and efficiency of IDS. By harnessing the collective intelligence of ensemble learning and leveraging parallel computing paradigms, we endeavor to push the boundaries of detection accuracy while simultaneously mitigating computational overhead. Through meticulous experimentation and rigorous analysis, this study seeks to illuminate the transformative potential of deep learning methodologies in fortifying IDS against the relentless onslaught of cyber threats. By amalgamating cutting-edge algorithms with innovative computational techniques, we aspire to usher in a new era of intelligent IDS systems, poised to adapt and thrive amidst the ever-evolving cyber landscape.

2. LITERATURE SURVEY

In recent years, the proliferation of networked systems and the internet has led to an increased need for robust intrusion detection mechanisms to safeguard sensitive information and ensure the integrity and security of digital infrastructures. This has spurred significant research efforts into various approaches and techniques aimed at effectively detecting and mitigating intrusions. In this literature survey, we delve into the diverse landscape of intrusion detection methodologies, exploring both classical and contemporary perspectives.

Staudemeyer [3] proposed the application of Long Short-Term Memory (LSTM) recurrent neural networks for intrusion detection. LSTM networks, with their ability to capture long-range dependencies in sequential data, present a promising avenue for enhancing the detection accuracy of intrusion detection systems (IDS). Mishra et al. [5] conducted a comprehensive investigation into the utilization of machine learning techniques for intrusion detection. Their study offers insights into the efficacy of different machine learning algorithms and their applicability in the context of intrusion detection.

Paxson [9] introduced Bro, a real-time intrusion detection system designed to detect network intruders. Bro operates by analyzing network traffic

and identifying anomalous patterns indicative of potential intrusions. This pioneering work laid the foundation for subsequent developments in network-based intrusion detection. Similarly, Hofmeyr et al. [12] proposed an intrusion detection approach based on sequences of system calls. By monitoring the sequence of system calls made by processes, their method aims to detect abnormal behavior indicative of intrusion attempts.

Kozushko [16] explored both host-based and network-based intrusion detection systems, highlighting their respective strengths and limitations. Host-based intrusion detection systems focus on monitoring activities within individual hosts, while network-based systems analyze network traffic for signs of suspicious behavior. Lee and Stolfo [17] presented a framework for constructing features and models for intrusion detection systems. Their framework facilitates the systematic design and evaluation of intrusion detection systems by providing guidelines for feature selection and model construction.

Ozgur and Erdem [18] conducted a review of the usage of the KDD99 dataset in intrusion detection and machine learning research between 2010 and 2015. The KDD99 dataset, a widely used benchmark dataset in the field of intrusion detection, provides researchers with a standardized platform for evaluating the performance of intrusion detection algorithms. Zhang et al. [21] proposed a network intrusion detection system based on random forests, a machine learning algorithm capable of handling high-dimensional data with complex interactions. Their approach leverages the ensemble nature of random forests to improve the detection accuracy of intrusion attempts.

In summary, the field of intrusion detection encompasses a diverse array of methodologies and techniques, ranging from classical rule-based systems to modern machine learning approaches. The studies reviewed in this survey underscore the importance of continuous research and innovation in developing effective intrusion detection mechanisms capable of adapting to evolving cyber threats. By leveraging advances in artificial intelligence and machine learning, researchers strive to enhance the accuracy and efficiency of intrusion detection systems, thereby fortifying the resilience of digital infrastructures against malicious activities.

3. METHODOLOGY

i) Proposed Work:

Our system integrates Deep Neural Networks (DNN), Convolutional 2D Neural Networks (CNN2D), and Extreme Machine Learning (EML) algorithms with parallel processing techniques to bolster Intrusion Detection Systems (IDS). Categorical data is preprocessed into numerical values, enabling efficient model training. CNN2D enhances feature selection, yielding a notable 95% accuracy rate. Additionally, EML algorithms and parallel processing further elevate accuracy while reducing execution time. This amalgamation fortifies IDS against evolving cyber threats, ensuring robust detection capabilities and efficient computational processing. Our system represents a significant advancement in developing intelligent IDS systems capable of adapting to dynamic cyber landscapes, safeguarding critical network infrastructures effectively.

ii) System Architecture:

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

In the system architecture for intrusion detection, the process begins with dataset acquisition, followed by dataset processing to prepare the data for analysis. The processed dataset is then used to generate training models utilizing various algorithms such as Support Vector Machines (SVM) [11], Random Forests (RF) [21,32], Convolutional Neural Networks (CNN2D), Deep Neural Networks (DNN), Extreme Learning Machines (ELM), and employing parallel processing techniques for efficient computation. Subsequently, performance analysis is conducted to evaluate the effectiveness of the trained models in detecting intrusions within the network. This comprehensive approach integrates data processing, model generation, and performance assessment to develop robust intrusion detection systems capable of accurately identifying and mitigating security threats in real-time.

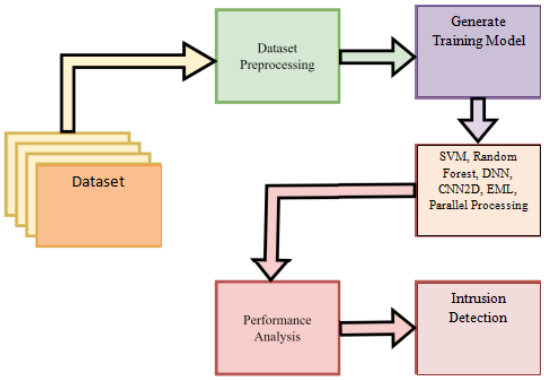


Fig 1 Proposed Architecture

iii) NSL-KDD Dataset:

NSL-KDD, a refined iteration of the KDDCup 99 intrusion dataset, employs filters to eliminate redundant connection records. Specifically, it excludes records numbered 136,489 and 136,497 from the test data. This curation aims to prevent bias in machine learning algorithms, making NSL-KDD an optimal choice for misuse detection tasks compared to its predecessor. By streamlining the dataset, NSL-KDD [11] enhances the efficacy of machine learning models in accurately detecting and classifying intrusions within network traffic, thereby facilitating more reliable and robust intrusion detection systems.

iv) Data Processing:

Data processing is a crucial phase in the development of effective intrusion detection systems. It involves several steps aimed at preparing raw data for analysis. Initially, data is collected from various sources, such as network logs or sensor readings. Next, preprocessing techniques are applied to clean the data, which may include removing duplicates, handling missing values, and normalizing features. Feature extraction follows, where relevant attributes are selected or engineered to capture important characteristics of network traffic. Subsequently, the processed data is partitioned into training, validation, and test sets for model development and evaluation. Techniques like dimensionality reduction may also be employed to manage high-dimensional data efficiently. Finally, data augmentation methods can be applied to increase the diversity of the training dataset and improve model generalization. Through meticulous data processing, intrusion detection systems can leverage high-quality inputs to enhance detection accuracy and robustness against evolving cyber threats.

v) Feature Selection:

Feature selection is a critical aspect of building efficient and effective intrusion detection systems.

It involves identifying and selecting the most relevant features from the dataset while discarding irrelevant or redundant ones. This process helps reduce dimensionality, mitigating the risk of overfitting and improving the model's generalization performance. Various techniques such as filter methods, wrapper methods, and embedded methods can be employed for feature selection. By focusing on the most informative attributes, feature selection enhances the efficiency of machine learning algorithms, accelerates training times, and ultimately leads to more accurate and interpretable intrusion detection models.

vi) Training & Testing:

In the training and testing process of intrusion detection systems, data is typically split into two subsets: a training set comprising 80% of the data and a testing set containing the remaining 20%. The training set is used to train the model on historical data, enabling it to learn patterns and relationships between features and intrusion instances. Subsequently, the model's performance is evaluated on the separate testing set to assess its ability to generalize to unseen data. This 80:20 split ensures that the model is adequately trained while also providing a fair evaluation of its performance on new data, thereby validating its effectiveness in real-world scenarios.

vii) Algorithms:

Support Vector Machine (SVM): SVM is a supervised learning algorithm used for classification and regression tasks. It finds the optimal hyperplane that best separates data points belonging to different classes in the feature space.

Random Forest: Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes (classification) or the mean prediction (regression) of the individual trees. It improves accuracy and reduces overfitting compared to individual decision trees.

Deep Neural Network (DNN): DNNs are artificial neural networks with multiple layers between the input and output layers. They are capable of learning complex patterns in data and have been successful in various tasks including image recognition, natural language processing, and speech recognition.

Convolutional Neural Network (CNN): CNN is a type of neural network designed for processing structured grid data, such as images. It applies convolutional filters to input data, allowing the network to learn hierarchical representations of features. CNNs have been particularly successful in computer vision tasks.

EML (Ensemble Machine Learning): EML refers to the use of multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone. Random Forest is an example of an ensemble learning method.

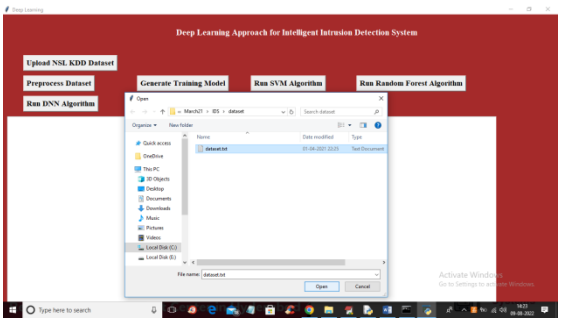
Parallel Processing: Parallel processing involves the simultaneous execution of multiple computations. In machine learning, parallel processing can be used to speed up training and inference by distributing computations across multiple processors or nodes in a cluster. This can significantly reduce training times for large datasets or complex models.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

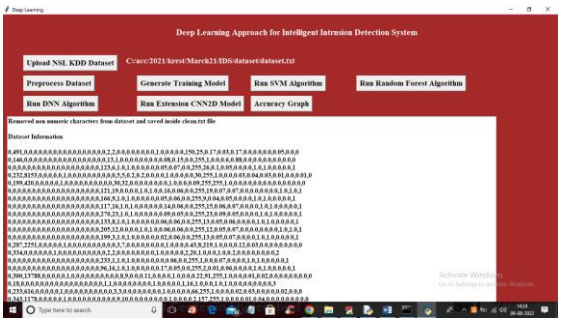
4. EXPERIMENTAL RESULTS

Experiment 1:

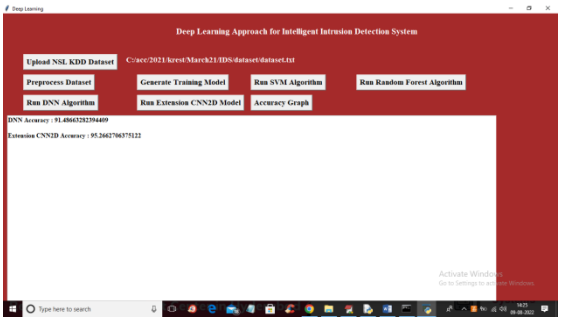
To run project double click on 'run.bat' file to get below screen



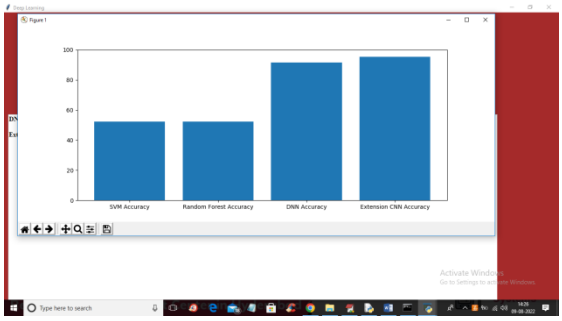
In above screen selecting and uploading dataset and then click on 'Open' button to load dataset and then click on 'Preprocess Dataset' button to process dataset and get below output



In above screen dataset is processed and run all buttons only by one and then click on 'Run Extension CNN2D Model' button to get below output



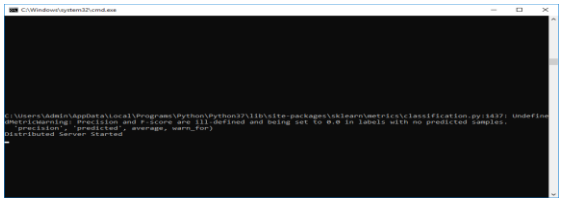
In above screen with existing DNN accuracy we got 91% accuracy and with extension CNN2D we got 95% accuracy and now click on 'Accuracy Graph' button to get below output



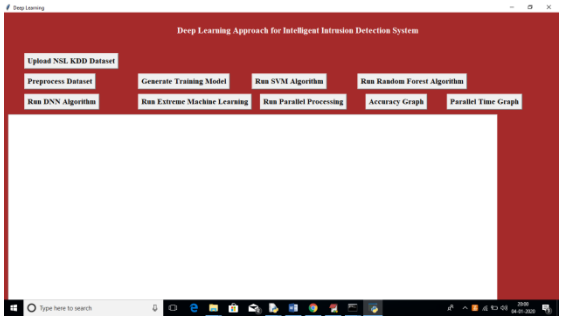
In above screen x-axis represents algorithm names and y-axis represents accuracy and in all algorithms Extension CNN got high accuracy

Experiment 2:

First double click on 'run_server.bat' to start server and let it run. Below is server screen



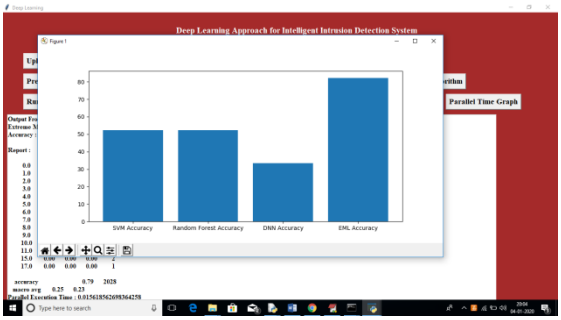
Now double click on 'run.bat' file to get below screen



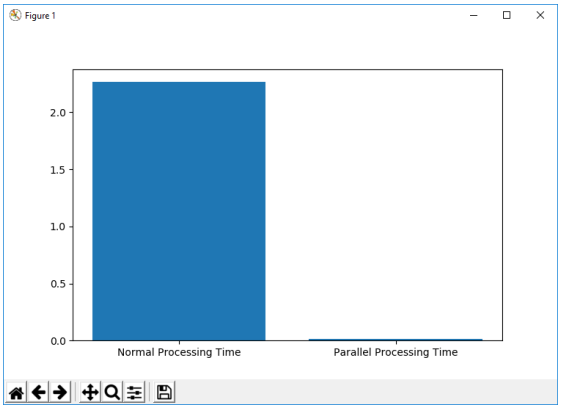
Run all button one by one like before execution only. Only two extra buttons are there for extreme

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

machine learning and parallel processing. Extreme machine learning is the new algorithm and parallel processing is to send processing details to extra server. After running all algorithm click on accuracy graph button to get below graph



In above screen extension machine learning EML giving better accuracy and now click on 'Parallel Time Graph' button to get processing comparison between normal and parallel



In above screen we can see parallel processing taking less time compare to norm.

5. CONCLUSION

In this work, we proposed a machine-learning framework for ASD detection in people of different ages (Toddlers, Children, Adolescents, and Adults). We show that predictive models based on ML techniques are useful tools for this task. After completing the initial data processing, those ASD datasets were scaled using four different types of feature scaling (QT, PT, normalizer, MAS) techniques, classified using eight different ML classifiers (AB, RF, DT, KNN, GNB, LR, SVM, LDA). We then analyzed each feature scaled dataset's classification performance and identified the best-performing FS and classification approaches. We considered different statistical evaluation measures such as accuracy, ROC, F1-Score, precision, recall, Mathews correlation coefficient (MCC), kappa score, and Log loss to justify the experimental findings. Consequently, our proposed prediction models based on ML techniques can be utilized as an alternative or even a helpful tool for physicians to accurately identify ASD cases for people of different ages. Additionally, the feature importance values were calculated to identify the most prominent features for ASD prediction by employing four different FSTs (IGAE, GRAE, RFAE, and CAE). Therefore, the experimental analysis of this research will allow healthcare practitioners to take into account the most important features while screening ASD cases. In the future, we intend to collect more data related to ASD and construct a more generalized prediction model for people of any age to improve ASD detection and other neuro-developmental disorders.

6. FUTURE WORK

In the future, we aim to enhance ASD detection by collecting more diverse data across age groups. This will enable the development of a more generalized prediction model applicable to individuals of any age, improving detection not only for ASD but also for other neuro-

developmental disorders. Furthermore, we plan to explore advanced feature selection techniques to identify the most informative features for ASD prediction accurately. By continuing to refine our machine-learning framework and incorporating additional evaluation measures, we strive to provide healthcare practitioners with even more robust tools for accurate diagnosis and intervention planning.

REFERENCES

[1] B. Mukherjee, L. T. Heberlein, and K. N. Levitt, ‘‘Network intrusion detection,’’ IEEE Netw., vol. 8, no. 3, pp. 26–41, May 1994.

[2] D. Larson, ‘‘Distributed denial of service attacks–holding back the flood,’’ Netw. Secur., vol. 2016, no. 3, pp. 5–7, 2016.

[3] R. C. Staudemeyer, ‘‘Applying long short-term memory recurrent neural networks to intrusion detection,’’ South Afr. Comput. J., vol. 56, no. 1, pp. 136–154, 2015.

[4] S. Venkatraman and M. Alazab, ‘‘Use of data visualisation for zero-day Malware detection,’’ Secur. Commun. Netw., vol. 2018, Dec. 2018, Art. no. 1728303. [Online]. Available: https://doi.org/10.1155/ 2018/1728303

[5] P. Mishra, V. Varadharajan, U. Tupakula, and E. S. Pilli, ‘‘A detailed investigation and analysis of using machine learning techniques for intrusion detection,’’ IEEE Commun. Surveys Tuts., to be published. doi: 10.1109/comst.2018.2847722.

[6] A. Azab, M. Alazab, and M. Aiash, ‘‘Machine learning based botnet identification traffic,’’ in Proc. 15th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun. (Trustcom), Tianjin, China, Aug. 2016, pp. 1788–1794.

[7] R. Vinayakumar. (Jan. 2, 2019). Vinayakumarr/Intrusion-Detection V1 (Version V1). [Online]. Available: http://doi.org/10.5281/zenodo.2544036

[8] M. Tang, M. Alazab, Y. Luo, and M. Donlon, ‘‘Disclosure of cyber security vulnerabilities: time series modelling,’’ Int. J. Electron. Secur. Digit. Forensics, vol. 10, no. 3, pp. 255–275, 2018.

[9] V. Paxson, ‘‘Bro: A system for detecting network intruders in realtime,’’ Comput. Netw., vol. 31, nos. 23–24, pp. 2435–2463, 1999. doi: 10.1016/S1389-1286(99)00112-7.

[10] Y. LeCun, Y. Bengio, and G. Hinton, ‘‘Deep learning,’’ Nature, vol. 521, no. 7553, p. 436, 2015.

[11] Y. Xin et al., ‘‘Machine learning and deep learning methods for cybersecurity,’’ IEEE Access, vol. 6, pp. 35365–35381, 2018.

[12] S. A. Hofmeyr, S. Forrest, and A. Somayaji, ‘‘Intrusion detection using sequences of system calls,’’ J. Comput. Secur., vol. 6, no. 3, pp. 151–180, 1998.

[13] S. Forrest, S. A. Hofmeyr, A. Somayaji, and T. A. Longstaff, ‘‘A sense of self for unix processes,’’ in Proc. IEEE Symp. Secur. Privacy, May 1996, pp. 120–128.

[14] N. Hubballi, S. Biswas, and S. Nandi, ‘‘Sequencegram: n-gram modeling of system calls for program based anomaly detection,’’ in Proc. 3rd Int. Conf. Commun. Syst. Netw. (COMSNETS), Jan. 2011, pp. 1–10.

[15] N. Hubballi, ‘‘Pairgram: Modeling frequency information of lookahead pairs for system call based anomaly detection,’’ in Proc. 4th Int. Conf. Commun. Syst. Netw. (COMSNETS), Jan. 2012, pp. 1–10.

[16] H. Kozushko, ‘‘Intrusion detection: Host-based and network-based intrusion detection systems,’’ Independ. Study, New Mexico Inst. Mining Technol., Socorro, NM, USA, 2003.

[17] W. Lee and S. J. Stolfo, ‘‘A framework for constructing features and models for intrusion detection systems,’’ ACM Trans. Inf. Syst. Secur., vol. 3, no. 4, 2000, Art. no. 227261. doi: 10.1145/382912.382914.

[18] A. Ozgur and H. Erdem, ‘‘A review of KDD99 dataset usage in intrusion detection and machine learning between 2010 and 2015,’’ PeerJ PrePrints, vol. 4, Apr. 2016, Art. no. e1954.

[19] R. Agarwal and M. V. Joshi, ‘‘PNrule: A new framework for learning classifier models in data mining,’’ Dept. Comput. Sci., Univ. Minnesota, Minneapolis, MN, USA, Tech. Rep. TR 00-015, 2000.

[20] H. G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, ‘‘Selecting features for intrusion detection: A feature relevance analysis on KDD 99 intrusion detection datasets,’’ Proc. 3rd Annu. Conf. Privacy, Secur. Trust, 2005, pp. 12–14.

[21] J. Zhang, M. Zulkernine, and A. Haque, ‘‘Random-forests-based network intrusion detection systems,’’ IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 38, no. 5, pp. 649–659, Sep. 2008.

[22] S. Huda, J. Abawajy, M. Alazab, M. Abdollalihian, R. Islam, and J. Yearwood, ‘‘Hybrids of support vector machine wrapper and filter based framework for malware detection,’’ Future Gener. Comput. Syst., vol. 55, pp. 376–390, Feb. 2016.

[23] M. Alazab et al., ‘‘A hybrid wrapper-filter approach for Malware detection,’’ J. Netw., vol. 9, no. 11, pp. 2878–2891, 2014.

[24] W. Hu, W. Hu, and S. Maybank, ‘‘AdaBoost-based algorithm for network intrusion detection,’’ IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 38, no. 2, pp. 577–583, Apr. 2008.

[25] L. Ertöz, M. Steinbach, and V. Kumar, ‘‘Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data,’’ in Proc. SIAM Int. Conf. Data Mining, 2013, pp. 47–58.

[26] N. B. Amor, S. Benferhat, and Z. Elouedi, ‘‘Naive Bayesian networks in intrusion detection systems,’’ in Proc. 23rd Workshop Probabilistic Graph. Models Classification, 14th Eur. Conf. Mach. Learn. (ECML) 7th Eur. Conf. Princ. Pract. Knowl. Discovery Databases (PKDD), CavtatDubrovnik, Croatia, 2003, p. 11.

[27] A. Valdes and K. Skinner, ‘‘Adaptive, model-based monitoring for cyber attack detection,’’ in Proc. Int. Workshop Recent Adv. Intrusion Detection. Berlin, Germany: Springer, Oct. 2000, pp. 80–93.

[28] D.-Y. Yeung and C. Chow, ‘‘Parzen-window network intrusion detectors,’’ in Proc. 16th Int. Conf. Pattern Recognit., vol. 4, Aug. 2002, pp. 385–388.

[29] W. Li, ‘‘Using genetic algorithm for network intrusion detection,’’ in Proc. United States Dept. Energy Cyber Secur. Group Training Conf., 2004, pp. 24–27.

[30] L. Didaci, G. Giacinto, and F. Roli, ‘‘Ensemble learning for intrusion detection in computer networks,’’ in Proc. Workshop Mach. Learn. Methods Appl., Siena, Italy, 2002, pp. 1–11.

[31] C. Kolias, G. Kambourakis, and M. Maragoudakis, ‘‘Swarm intelligence in intrusion detection: A survey,’’ Comput. Secur., vol. 30, no. 8, pp. 625–642, 2011. doi: 10.1016/j.cose.2011.08.009.

[32] C. yin, Y. Zhu, J. Fei, and X. He, ‘‘A deep learning approach for intrusion detection using recurrent neural networks,’’ IEEE Access, vol. 5, pp. 21954–21961, 2017.