Mugla Journal of Science and Technology

APPLICATION OF EXPLAINABLE ARTIFICIAL INTELLIGENCE IN INTRUSION DETECTION SYSTEM CLASSIFICATION USING BOOSTING ALGORITHMS

Ercan ATAGÜN*, Department of information security and management, OSYM, TÜRKİYE, ercan.atagun@osym.gov.tr
(https://orcid.org/0000-0001-5196-5732)
Günay TEMÜR, Kaynasli Vocational School, Düzce University, TÜRKİYE, gunaytemur@duzce.edu.tr
(https://orcid.org/0000-0002-7197-5804)
Serdar BİROĞUL, Computer Engineering Department, Düzce University,TÜRKİYE, serdarbirogul@duzce.edu.tr
(https://orcid.org/0000-0003-4966-5970)

Received: 14.08.2023, Accepted: 13.12.2023 Research Article
*Corresponding author DOI: 10.22531/muglajsci.1343051

Abstract

The increased speed rates and ease of access to the Internet increase the availability of devices with Internet connections. Internet users can access many devices that they are authorized or not authorized. These systems, which detect whether users have unauthorized access or not, are called Intrusion Detection Systems. With intrusion detection systems, users' access is classified and it is determined whether it is a normal login or an anomaly. Machine learning methods undertake this classification task. In particular, Boosting algorithms stand out with their high classification performance. It has been observed that the Gradient Boosting algorithm provides remarkable classification performance when compared to other methods proposed for the Intrusion Detection Systems problem. Using the Python programming language, estimation was made with the Gradient Boost, Adaboost algorithms, Catboost, and Decision Tree and then the model was explained with SHAPASH. The goal of SHAPASH is to enable universal interpretation and comprehension of machine learning models. Providing an interpretable and explainable approach to Intrusion Detection Systems contributes to taking important precautions in the field of cyber security. In this study, classification was made using Boosting algorithms, and the estimation model created with SHAPASH, which is one of the Explainable Artificial Intelligence approaches, is explained.

Keywords: Intrusion detection system, Explainable artificial intelligence, Gradient boosting

BOOSTING ALGORİTMALARI KULLANARAK SALDIRI TESPİT SİSTEMLERİ SINIFLANDIRMADA AÇIKLANABİLİR YAPAY ZEKA UYGULAMASI

Özet

İnternete erişimin kolaylaşması ve hız oranlarının artması ile birlikte internete bağlı cihazlara erişimi de arttırmaktadır. İnternet kullanıcıları yetkili oldukları veya yetkilendirilmedikleri birçok cihaza erişebilirler. Kullanıcıların yetkisiz erişime sahip olup olmadığını tespit eden bu sistemlere Saldırı Tespit Sistemleri denir. Saldırı tespit sistemleri ile kullanıcıların erişimleri sınıflandırılır ve normal bir giriş mi yoksa bir anormallik mi olduğu belirlenir. Makine öğrenimi yöntemleri bu sınıflandırma görevini üstlenir. Özellikle Boosting algoritmaları, yüksek sınıflandırma performansları ile öne çıkmaktadır. Gradient Boosting algoritmasının Saldırı Tespit Sistemleri problemi için önerilen diğer yöntemlere göre dikkate değer bir sınıflandırma performansı sağladığı gözlemlenmiştir. Python programlama dili kullanılarak Gradient Boost ve Adaboost algoritmaları ile tahmin yapılmış ve ardından model SHAPASH ile açıklanmıştır. SHAPASH, makine öğrenmesi modellerinin herkes tarafından yorumlanabilir ve anlaşılır hale getirmeyi hedeflemektedir. Saldırı Tespit Sistemleri için yorumlanabilir ve açıklanabilir bir yaklaşım sunulması siber güvenlik alanında önemli tedbirlerin alınmasında katkı sağlamaktadır. Bu çalışmada Boosting algoritmaları kullanılarak sınıflandırma yapılmış ve Açıklanabilir Yapay Zeka yaklaşımlarından biri olan SHAPASH ile oluşturulan tahmin modeli anlatılmıştır.

Anahtar Kelimeler: İzinsiz giriş tespit sistemi, Açıklanabilir yapay zeka, Gradient boosting

Cite

Atagün, E., Temür, G., Biroğul, S., (2024). "Applications of Explainable Artificial Intelligence in Intrusion Detection System Classification Using Boosting Algorithm", Mugla Journal of Science and Technology, 10(1), 1-7.

1. Introduction

The rapid development of internet technologies has brought about an increase in internet access. Internet access has enabled many users to access Internet devices. It is also possible to increase authorized or unauthorized access to internet devices. Detection of intruders is extremely important for safe internet use.

The process of monitoring events occurring on a computer system [1] or computer network and

investigating possible incidents about acceptable use policies or security breaches is called Intrusion Detection System (IDS). Thanks to malicious software such as spyware and worms, unauthorized privileges can be obtained by making unauthorized attempts to systems on the internet.

Since the internet is widely used in industry and scientific applications, attacks on internet networks cause many negativities. Some of these negativities can be listed as inability to provide services, loss of financial income, loss of information assets, and damage to information integrity.

IDS is an essential part of network security systems used to detect attacks on a computer network or the Internet [2]. Detection mechanisms are examined in two groups signature-based and anomaly-based [3]. Signature-based IDSs are used in attack detection, and when a request is received, a comparison is made with this database. Thanks to this comparison, it is understood whether it is an anomaly or a normal request [3]. In anomaly-based systems, a statistical and machine learning-based model is established and thus it is determined whether a request is normal or anomaly.

Intrusion Detection Systems and Machine Learning applications on this subject have been evaluated in detail. This paper is organized as follows:

Section 2 presents work on various datasets, Section 3 explores material and method motivation, Section 4 presents empirical results of the problem, and Section 5 presents results and discusses future work.

2. Related Works

Researchers have been interested in cybersecurity attacks for a long time. Especially with the publication of the KDD data set in 1999 [4], especially artificial intelligence researchers have shown the necessary interest in this field.

Levin [5] developed Kernel Miner and attracted attention by winning second place in the KDD Classifier Learning Competition in 1999. Manzoor and Kumar [6] emphasized the need for preprocessing to compensate for attacks that occurred less and more and proposed a reduced IDS using Artificial Neural Network. Alzubi et al. [7] proposed a solution for IDS systems by hybridizing modified binary Gray Wolf Optimization with Particle Swarm Optimization. Abd Elaziz et al. [8] proposed a novel IDS approach based on deep learning and swarm intelligence (Capuchin Search Algorithm). Hussain and Lalmuanawma [9] applied the JRip and J48 methods to the IDS problem by focusing on the noisy data problem of the KDD dataset. They argued that noisy data would yield more realistic results. Ruan and Miao [10] stated that they lacked visualization for IDS, especially for the KDD99 dataset, and made data visualization.

They used the hash algorithm and treated the KDD99 dataset in terms of volume, speed, and diversity. Al Mehedi Hasan et al. [11] proposed the Support Vector Machine to provide solutions to the IDS problem and emphasized the importance of choosing the appropriate kernel and parameters of the Support Vector Machine algorithm. Kandeeban & Rajesh [12] provided the classification of intrusion detections by including the Genetic Algorithm in IDS. Rule sequences were derived from the training dataset reduced by the Genetic Algorithm and the fitness function was defined to evaluate the quality of the rules. Nuiaa et al. [13] proposed the Evolving Dynamic Fuzzy Clustering model for attack detection systems.

Sahu et al. [14] dealt with preprocessing steps such as completing missing data, removing redundant data, normalization, size reduction, and selecting the most relevant features on the scale of the IDS problem. They used KNN, Support Vector Machine, K-Mean, Decision Tree, and Fuzzy C-Mean Clustering algorithms. Tavallaee et al. [15] presented a statistical analysis of the KDD99 dataset for IDS. Shone et al. [16] suggested a nonsymmetric deep autoencoder (NDAE) and a Deep Learnig approach for IDS. Niu et al. [17] suggested multi-part feature generation and XGBoost method for the IDS problem. Ingre and Yadav [18] performed a performance analysis for the NSL-KDD dataset used for the IDS problem using Artificial Neural Network. Ambusaidi et al. [19] proposed an algorithm that aims to determine the most suitable feature for estimation in the IDS problem. They aimed for both higher accuracy and lower computational cost with the method they named an IDS based on the Least Squares Support Vector Machine (LSSVM-IDS). Ferrag et al. [20] present a study of deep learning approaches and used CSE-CIC-IDS2018 datasets for IDS.

Beechey et al. [21] used a Decision Tree, to protect networked systems from cyber threats and achieved an F1 score of 0.99. Explainable Artificial Intelligence (XAI) approaches for IDS provide interpretable [22] in cyber security studies. By proposing a Deep learning-based model for IDS, LIME, and SHAP explainable artificial intelligence approaches were applied [23]. Wang et al. [24], there is a study that makes explanations in IDSs with the SHAP method using the NSL-KDD dataset. Mallampati and Seetha [25] provide a review of IDSs in the context of machine learning, deep learning, and Explainable Artificial Intelligence. Patil et al. [26] improved the classification performance for IDS using the CICIDS-2017 dataset and applied it with LIME. Kharwar and Thakor[27] proposed a hybrid model combining sequential forward/backward floating selection with extra-tree and the XGBoost algorithm for the IDS problem.

Carrera et al. [28] aims to detect the most important element when predicting whether a transaction is an attack or not by using Shap in the IDS problem. Sivamohan and Sridhar, [29] presented a cyber threat sensitive solution to the IDS problem with a BiLSTM-XAI based approach.

3. Material and Method

In the intrusion detection systems data set, there are 3 different categorical values in the attribute column named protocol type. These are Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Internet Control Message Protocol (ICMP).

ICMP is among the Network layer protocols in the Open Systems Interconnection (OSI) reference model. While TCP continues processing by receiving the ACK (acknowledgment) packet, UDP continues the process without the need for ack acknowledgment. In this regard, while TCP provides reliable data transfer, UDP provides unreliable data transfer. ICMP is a protocol that runs at the Network layer in the OSI reference model. ICMP, Time-to-live notify the owner of the expired package, feedback about the destroyed package undertakes such tasks.

One of the most important processes to determine the classification performance of Intrusion Detection Systems is to obtain the appropriate data set. In this study, KDD-99 dataset [4], which is widely used in intrusion detection systems, was used to solve the classification problem and subsequently apply explainable artificial intelligence. This data set was developed for the testing of Intrusion detection systems and was prepared and made available within the framework of the 5th International Knowledge Discovery and Data Mining conference. Building a network intrusion detector—a prediction model that can tell the difference between "good" normal connections and "bad" connections known as intrusions or attacks—was the goal for the conference. There are 41 attributes and 3 of them have categorical values: protocol_type, service, flag. Since there is no missing value in the data set, there was no need for data preprocessing steps such as creating data with mean, median or deleting data. The protocol_type feature takes tcp, udp and icmp values. While the feature named service receives many categorical values, the most common service types are: private, http, telnet, smtp, ecr_i.

3.1. Gradient Boosting

The Gradient Boosting algorithm is one of the most powerful ML algorithms used in many different applications. It stands out with the operations above the loss function and the customization of different loss functions to a certain extent according to the problem needs [30].

The Gradient Boosting algorithm creates an ML model by combining weak prediction models to form a decision tree. The loss function is meant to be minimized. Boosting means correcting the error of each estimator. In this regard, the algorithm plans to increase its performance with the help of the previous estimator's error.

3.2. Adaboost

Adaboost algorithm, also known as Adaptive Boosting, is an Ensemble Learning algorithm that makes class label prediction with the help of a group of weak classifiers. Adaboost, weak classifiers constitute the decision mechanism. A classifier that provides a higher classification [31] is obtained by combining weak classifiers. Although Adaboost is similar to the Random Forest algorithm in the process of determining each decision tree in the model, it restricts the tree size. Adaboost algorithm regulates the misclassification [32] in the model together with other models and thanks to this feature, its use has become widespread in recent years.

3.3. CatBoost

Catboost algorithm[33] is a machine learning algorithm developed in 2017 and is based on Gradient Boosting. Catboost has its own coding method during the data preparation phase. Therefore, it does not need encoding. Catboost aims to increase prediction performance by creating symmetric trees.

3.4. Decision Tree

Decision Tree is one of the supervised learning algorithms used in both classification and regression problems. Entropy calculates the homogeneity of a feature [34] and if the samples are divided equally, the entropy value is equal to one. When creating the decision tree, the best divisor attribute is selected. This feature is converted into a decision node. Afterwards, the data set is divided into further sub-sets.

3.5. Shapash

Shapash is written in Python and makes machine learning methods explainable and interpretable. Shapash offers a variety of visualizations that use clear labeling that anyone can comprehend [35]. With Shapash, clear, understandable results are displayed [35] and also the dataset and models become interpretable. It is easy to use with a live and interactive interface. With this interactive interface, users can easily discuss and interpret the problem. It is used to determine the effect of the relative relationship between the value estimated by Shapash and the variables [36]. Shapash effectively produces results in regression problems, binary classification, or multiple classification problems. Shapash works in harmony with models such as Ensemble models, Catboost, Xgboost, LightGBM, Linear models, and SVM [37] in the current updates.

4. Experimental Results

In the study, the data were separated as 80% training and 20% testing, and experimental results were obtained. The target variable defined as 'normal' and 'anomaly' in the data set was tried to be estimated. It showed 99.44% classification success with the GBM algorithm.

Table 1 shows classification metrics.

Table 1. IDS Classification Metrics.

Algorithm	Precision	Recall	Accuracy	F1 Score
Gradient Boosting	0.9953	0.9963	0.9944	0.996
Adaboost	0.9937	0.9937	0.9942	0.995
Catboost	0.9798	0.9843	0.9809	0.982
Decision Tree	0.9798	0.9927	0.9853	0.986

Figure 1 shows the complexity matrix of the Accuracy and Gradient Boosting algorithm.

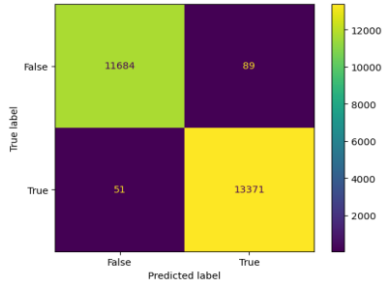

Figure 1. Gradient Boosting Confusion Matrix.

Figure 2 shows Gradient Boosting with Shapash the analysis of data with the class label 'Normal'.

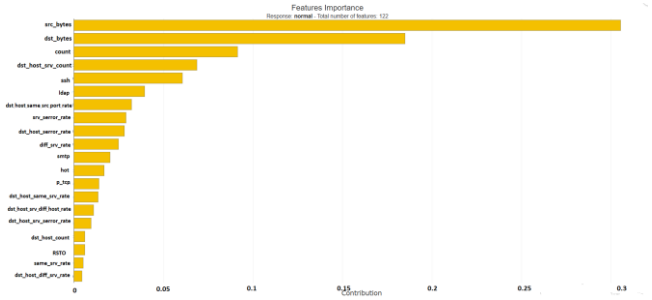

Figure 2. GB with KDD Dataset Shapash with Target Variable 'Normal'.

When Figure 2 is examined, the most important attributes with the highest Feature Importance value are src_bytes, dst_bytes, and count. Of these attributes, src_bytes has the highest Feature Importance value of 0.3053. From other attributes dst_bytes 0.1849, the count has a value of 0.091. Minimum Feature Importance values are dst_hot_diff_srv_rate, same srv_rate, and RSTO. RSTO has the lowest Feature Importance with 0.0061 while dst_hot_diff_srv_rate is 0.0045 and the same srv_rate: is 0.0052.

Figure 3 shows the Features Importance obtained by Shapash and Catboost algorithm. Compared to Figure 2, src_bytes, dst_bytes, count are in the first 3 places. Gradient Boosting's fourth ranked attribute is dst_host_srv_count, while Catboost's fourth ranked attribute is ssh.

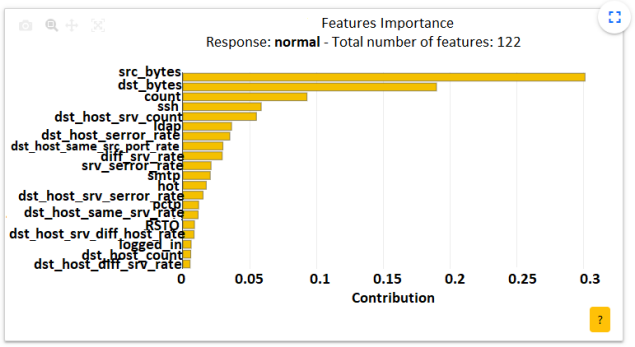

Figure 3. Catboost with Local Explanation and 'anomaly'.

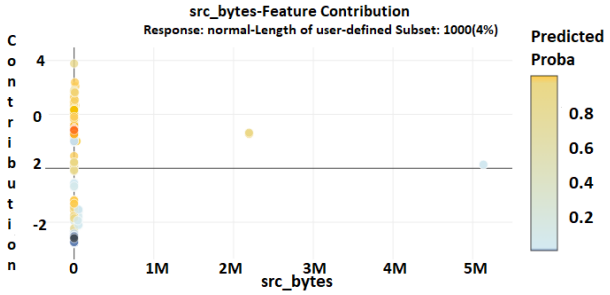

Figure 4. Feature Contribution.

When Figure 4 is examined, a contribution of 3.8852 is provided for the "Normal" class, thanks to the value with the src_bytes value of at least 30, and the model estimated the data in question as "normal" thanks to the significant contribution of this value. When the src_bytes variable is examined for the highest value of 5133876, its contribution to the model is -0.1394. It has been observed that the model predicts 'anomaly' for the data handled with this - statement. As it can be understood from here, the high value for src_bytes does not indicate that it contributes to the model. When another value is examined, the contribution to the model is observed as -2.7814 while the value of src_bytes is 0.

When Figure 5 was examined, the highest value for the Anomaly class was 1, while it was observed as 2.7814. Compared to the previous Figure X, Figure 6 and Figure 7 overlap since it is observed that the variable that contributes the least to the model for the Normal class is the src_bytes value of 1, and its contribution to the model is -2.7814. In this respect, it is very important which target variable is chosen. Similarly, when the src_bytes variable is examined for the highest value of 5133876, its contribution to the model is 0.1394. While this value was -0.1394 in Figure 4, it was 0.1394 in Figure 5. The reason for this is that the target variable was the anomaly selected in Figure 5 and the importance of this variable was determined in this way by Shapash Anomaly Class Feature Contribution is shown in Figure 5.

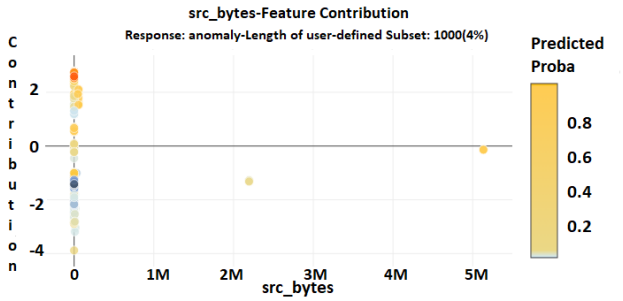

Figure 5. Feature Contribution with 'src_bytes'.

Figure 6 shows the Feature Contribution obtained with Decision Tree.

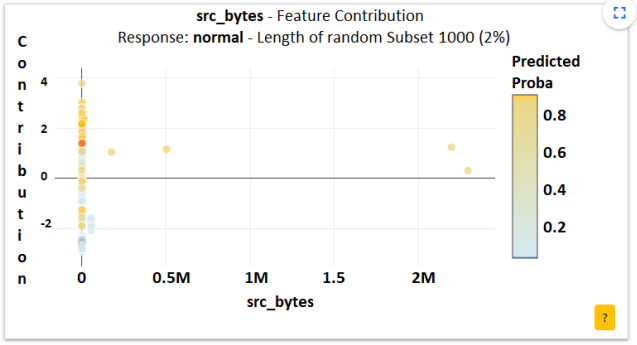

Figure 6. Decision Tree with Feature Contribution.

Figure 7 shows how to decide whether a data is normal or anomaly in an IDS problem using Decision Tree.

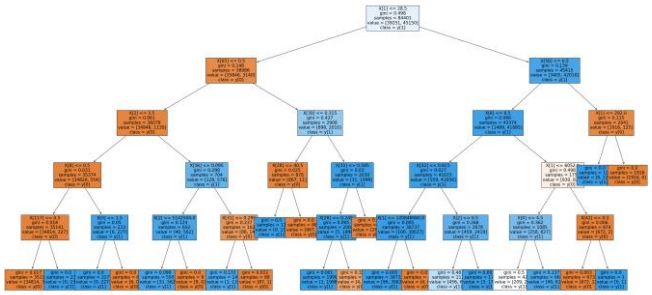

Figure 7. Decision Tree with sample data.

When Figure 8 is examined, the RSTO value provides the biggest contribution to the model. However, variables such as src_bytes, dst_bytes, ssh, count dst_host_srv_count, srv_serror_rate, dst_host_serror_rate, and diff_srv_rate made a negative contribution to the normal class by pointing out that the current data is an anomaly.

Figure 8 Feature Local Explanation is shown.

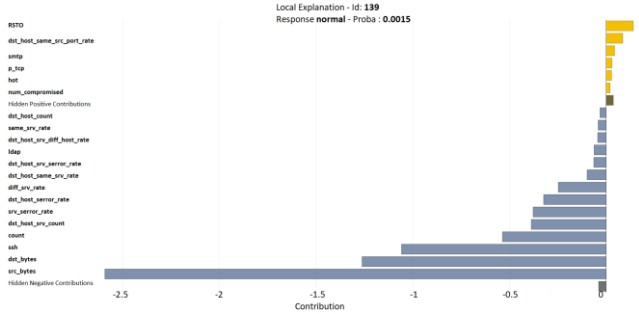

Figure 8. Feature Local Explanation.

Figure 9 is shown for Local Explanation and the 'anomaly' class.

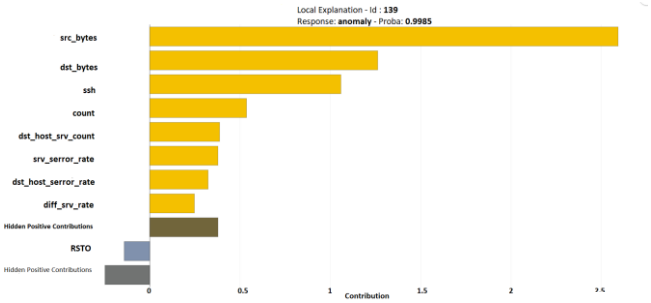

Figure 9. Local Explanation and 'anomaly'.

When Figure 9 is examined, it is estimated anomaly with the help of variables such as src_bytes, dst_bytes, ssh, count dst_host_srv_count, srv_serror_rate, dst_host_serror_rate, diff_srv_rate. However, the RSTO variable provides both a low contribution and a negative contribution to the estimation of the model. Src_bytes is a continuous and numeric variable that represents the number of bytes from source to destination. RSTO: Connection established, originator aborted (sent a RST). RSTO is an attribute derived from the flag attribute. There are 1562 RSTO values. The maximum value of RSTO is 1379963888. Its mean value is 45566.743 and its minimum value is 0. The standard deviation value is 5870331.182. RSTO value is distinct 3341. The RSTO value helped the classification algorithm in terms of explainability.

With Table 2, the id value in the data set is 139, while the src bytes value is 0, and the status of belonging to the anomaly class is determined as 0.9985 with Shapash. When the src bytes value is 0, they contributed 2.5937 to the model, when dst bytes was 0, 1.2623, when ssh was 1.0578, RSTO -0.1409 contributed.

Table 2 shows the examination of a data sample for the class label.

Table 2. Evaluation Of A Data Sample.

id	src bytes	dst bytes	ssh	RSTO	normal	anomaly
139	0	0	1.0578	-0.1409	0.0015	0.9985

5. Conclusion

The IDS problem discussed in this study, the proposed solution method, and the explainable approach presented for the prediction model, are aimed to be an inspiration for determining the best method for detecting future cyber attacks. Making cyber security measures interpretable with explainable artificial intelligence will remain on the agenda as an important study topic. Thus, XAI methods developed with machine learning methods in Internet of Things networks are the determinants of important developments in the field of cyber security. Making the lowest attributes useful in data preprocessing steps or when implementing classification algorithms should be considered. These features can be found in data reduction or another feature, and through data merging, new feature generation can be made in a way that will increase the

model prediction performance. Removing these attributes before giving the model to the model or generating another useful attribute from the data set will pave the way for higher performance. In addition, the storage cost can be reduced by removing the features that are not useful in terms of estimation algorithms for those who address this problem and aim to obtain a more original data set again. Instead, determining and obtaining other attributes that define the problem will enable the production of more unique data sets and higher performance in intrusion detection systems.

6. References

[1] Liao, H. J., Lin, C. H. R., Lin, Y. C., and Tung, K. Y., "Intrusion detection system: A comprehensive review", *Journal of Network and Computer Applications*, 36 (1), 16-24, 2013.

[2] Sharma S. and Gupta R. K., "Intrusion detection system: A review", *International Journal of Security and Its Applications*, 9 (5), 69–76, 2015.

[3] Özgür, A., and Erdem, H., "Saldırı tespit sistemlerinde genetik algoritma kullanarak nitelik seçimi ve çoklu sınıflandırıcı füzyonu", *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi,* 33(1), 75-87, 2018.

[4] Salvatore Stolfo, 2019. [Online]. Available: https://kdd.ics.uci.edu/databases/kddcup99/task.html. [Accessed 12 1 2023].

[5] Levin, I. "KDD-99 classifier learning contest LLSoft's results overview", *ACM SIGKDD Explorations Newsletter*, 1 (2), 67-75, 2000.

[6] Manzoor, I., and Kumar, N. "A feature reduced intrusion detection system using ANN classifier", *Expert Systems with Applications*, 88, 249-257, 2017.

[7] Alzubi, Q. M., Anbar, M., Sanjalawe, Y., Al-Betar, M. A., & Abdullah, R. "Intrusion detection system based on hybridizing a modified binary grey wolf optimization and particle swarm optimization", *Expert Systems with Applications*, 204, 117-597, 2022.

[8] Abd Elaziz, M., Al-qaness, M. A., Dahou, A., Ibrahim, R. A., and Abd El-Latif, A. A., "Intrusion detection approach for cloud and IoT environments using deep learning and Capuchin Search Algorithm", *Advances in Engineering Software*, 176, 103-402, 2023.

[9] Hussain, J., and Lalmuanawma, S., "Feature analysis, evaluation and comparisons of classification algorithms based on noisy intrusion dataset", *Procedia Computer Science*, 92, 188-198, 2016.

[10] Ruan, Z., Miao, Y., Pan, L., Patterson, N., and Zhang, J. "Visualization of big data security: a case study on the KDD99 cup data set", *Digital Communications and Networks*, 3 (4), 250-259, 2017.

[11] Al Mehedi Hasan, M., Nasser, M., and Pal, B., "On the KDD'99 dataset: support vector machine based intrusion detection system (ids) with different kernels", *International Journal of Electronics Communication and Computer Engineering*, 4 (4), 1164-1170, 2013.

[12] Kandeeban, S. S., and Rajesh, R. S., "A Genetic Algorithm Based elucidation for improving Intrusion Detection through condensed feature set by KDD 99 data set", *Information and Knowledge Management,* 1 (1), 1-9, 2011.

[13] Nuiaa, R. R., Alsaeedi, A. H., Manickam, S., and Al-Shammary, D. E. J., "Evolving dynamic fuzzy clustering (EDFC) to enhance DRDoS_DNS attacks detection mechanism", *International Journal of Intelligent Engineering & Systems*, 15 (1), 509-519, 2022.

[14] Sahu, S. K., Sarangi, S., and Jena, S. K., "A detail analysis on intrusion detection datasets", *2014 IEEE international advance computing conference*, 1348-1353, 2014.

[15] Tavallaee, M., Bagheri, E., Lu, W., and Ghorbani, A. A. "A detailed analysis of the KDD CUP 99 data set", *IEEE symposium on computational intelligence for security and defense applications*, 1-6, 2009.

[16] Shone, N., Ngoc, T. N., Phai, V. D., and Shi, Q., "A deep learning approach to network intrusion detection", *IEEE transactions on emerging topics in computational intelligence*, 2, 41-50, 2018.

[17] Niu, Y., Chen, C., Zhang, X., Zhou, X., and Liu, H., "Application of a New Feature Generation Algorithm in Intrusion Detection System", *Wireless Communications and Mobile Computing*, 1, 1-17, 2022.

[18] Ingre, B., and Yadav, A., "Performance analysis of NSL-KDD dataset using ANN", *2015 international conference on signal processing and communication engineering systems,* 92-96, 2015.

[19] Ambusaidi, M. A., He, X., Nanda, P., and Tan, Z., "Building an intrusion detection system using a filter-based feature selection algorithm", *IEEE transactions on computers*, 65 (10), 2986-2998, 2016.

[20] Ferrag, M. A., Maglaras, L., Moschoyiannis, S., and Janicke, H., "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study", *Journal of Information Security and Applications*, 50, 102-419, 2020.

[21] Beechey, M., Kyriakopoulos, K. G., and Lambotharan, S., "Evidential classification and feature selection for cyber-threat hunting", *Knowledge-Based Systems*, 226, 107-120, 2021.

[22] Moustafa, N., Koroniotis, N., Keshk, M., Zomaya, A. Y., and Tari, Z., "Explainable Intrusion Detection for Cyber Defences in the Internet of Things: Opportunities and Solutions", *IEEE Communications Surveys & Tutorials*, 1, 1-17, 2023.

[23] Sevri, M., and Karacan, H., "Explainable Artificial Intelligence (XAI) for Deep Learning Based Intrusion Detection Systems", *In The International Conference on Artificial Intelligence and Applied Mathematics in Engineering*, 39-55, Cham: Springer International Publishing, 2022.

[24] Wang, M., Zheng, K., Yang, Y., and Wang, X., "An explainable machine learning framework for intrusion detection systems", *IEEE Access*, 8, 73127-73141, 2020.

[25] Mallampati, S. B., and Seetha, H., "A Review on Recent Approaches of Machine Learning, Deep

Learning, and Explainable Artificial Intelligence in Intrusion Detection Systems", *Majlesi Journal of Electrical Engineering*, 17(1), 29-54, 2023.

[26] Patil, S., Varadarajan, V., Mazhar, S. M., Sahibzada, A., Ahmed, N., Sinha, O., and Kotecha, K., "Explainable artificial intelligence for intrusion detection system", *Electronics*, 11(19), 30-79, 2022.

[27] Kharwar, A., & Thakor, D. (2023). A hybrid approach for feature selection using SFFS and SBFS with extra-tree and classification using XGBoost. International Journal of Ad Hoc and Ubiquitous Computing, 43(4), 191-205.

[28] Carrera, F., Dentamaro, V., Galantucci, S., Iannacone, A., Impedovo, D., & Pirlo, G. (2022). Combining unsupervised approaches for near real-time network traffic anomaly detection. Applied Sciences, 12(3), 1759.

[29] Sivamohan, S., & Sridhar, S. S. (2023). An optimized model for network intrusion detection systems in industry 4.0 using XAI based Bi-LSTM framework. Neural Computing and Applications, 35(15), 11459-11475.

[30] Alexey Natekin, "Gradient boosting machines, a tutorial", 2013. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnbot.2013.00021/full. [Accessed 14 11 2022].

[31] Ravipati, R. D., and Abualkibash, M., "Intrusion detection system classification using different machine learning algorithms on KDD-99 and NSL-KDD datasets-a review paper", *International Journal of Computer Science & Information Technology*, 11(3), 65-80, 2019.

[32] Güllü, M., Polat, H., and Çetin, A., "Author identification with chicken swarm optimization algorithm and adaboost approaches", *International Conference on Computer Science and Engineering*, 1-5, 2020.

[33] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. Advances in neural information processing systems, 31.

[34] Ravipati, R. D., & Abualkibash, M. (2019). Intrusion detection system classification using different machine learning algorithms on KDD-99 and NSL-KDD datasets-a review paper. International Journal of Computer Science & Information Technology (IJCSIT) 11.

[35] Anonymous , "Welcome to Shapash's documentation", 2020. [Online]. Available: https://shapash.readthedocs.io/en/latest/. [Accessed 24 1 2022].

[36] Amin, M. N., Salami, B. A., Zahid, M., Iqbal, M., Khan, K., Abu-Arab, A. M., and Jalal, F. E., "Investigating the Bond Strength of FRP Laminates with Concrete Using LIGHT GBM and SHAPASH Analysis", *Polymers*, 14 (21), 1-16, 2022.

[37] Bouche T., "Overview", 2022. [Online]. Available: https://github.com/MAIF/shapash. [Accessed 26 11, 2022].