IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 29 August 2022, accepted 7 October 2022, date of publication 17 October 2022, date of current version 4 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3215269

RESEARCH ARTICLE

Data Discretization and Decision Boundary Data Point Analysis for Unknown Attack Detection

GUN-YOON SHIN[1], DONG-WOOK KIM[1], AND MYUNG-MOOK HAN[2]
[1]Department of Computer Engineering, Gachon University, Sungnam-si 13120, South Korea
[2]Department of AI Software, Gachon University, Sungnam-si 13120, South Korea

Corresponding author: Myung-Mook Han (mmhan@gachon.ac.kr)

This work was supported in part by the Ministry of Science and Information and Communication Technology (ICT) (MSIT), South Korea, through the Information Technology Research Center (ITRC) Support Program supervised by the Institute for Information & Communications Technology Planning & Evaluation (IITP) under Grant IITP-2022-2020-0-01602; and in part by the National Research Foundation of Korea (NRF) Grant funded by the Korea Government, MSIT, under Grant 2022R1F1A1073375.

ABSTRACT Researchers have continuously sought effective ways to detect unknown (zero-day) cyberattacks in real time. Most current methods rely on pattern-recognition to identify known threats when they appear. Recently, machine learning anomaly detection tools that train a model on normal network data have been used to identify outliers representing unknown attacks. However, detecting unknown attacks is difficult because of a lack of information on unknown attacks, class imbalance in the data, or failure to accurately detect attacks with normal patterns. To overcome these problems, this study applied data discretization and decision-boundary data point analyses to scrutinize patterns near the thresholds of uncertainty. A novel discretization method was used to effectively train a model for the fuzzy c-means feature analysis of data points at the decision boundary, through which adversarial features were detected and classified based on their entropy. Consequently, it was possible to identify incorrectly detected attack data distributed near the model's decision boundary. The NSL-KDD dataset, which is commonly used to evaluate ML intrusion detection systems, was used to evaluate the proposed method. The results showed that our model successfully identified attacks at the decision boundary and that its performance can be improved through classification. In addition, after classification, it was confirmed that the accuracy of detecting DoS attacks improved by 5 to 7%, Probe by 7 to 10%, R2L by 4 to 7%, and U2R by 1 to 9%, compared with that of existing models.

INDEX TERMS Data discretization, decision boundary, fuzzy c-means, network anomaly detection, unknown attack.

I. INTRODUCTION

The frequency and number of cyberattacks are ever-increasing, and brand-new zero-day threats are the most dangerous, as their signatures have not been codified by intrusion detection systems (IDSs) [1]. Conventional signature methods are effective on known attacks, but they are not effective on unknown attacks. Hence, researchers are actively seeking new methods to ensure computer network security [2]. To detect zero-day attacks before users are harmed, anomaly detection techniques are preferred, as they sense outlying network patterns based on dataset containing normal features [3], and the general method lends itself to ML techniques [4], [5]. However, even state-of-the-art anomaly detection methods have problems [6] as follows:

- Anomaly detection based on statistical outliers relies on recognizing unknown patterns, which are theoretically infinite in nature, and it frequently results in false-positives and requires delicate training methods that can easily lead to local minima biases, which cause true positives to be ignored.
- Outlying data points and behavior patterns are extremely difficult to classify, because of their unlimited bounds. Hence, data imbalances are frequent, which lowers performance accuracy.

The associate editor coordinating the review of this manuscript and approving it for publication was Giovanni Pau.

- Batch training also degrades performance, making it necessary to analyze and classify every feature and apply preprocessing measures for each type, which again creates problems in the face of unbounded sets.

Numerous studies have been conducted to solve these problems using combinations of preprocessing methods and ML approaches. Lakhina et al. [7] applied principal component analysis (PCA) to handle features collectively, and consequently, the processing time of the learning model was reduced. PCA and fuzzy methods [8] have been used to reduce processing time and add better judgment to the results, respectively. However, most of the previously applied preprocessing methods applied the same process to all features instead of applying processes suitable for each one. None of these methods accurately reflect the unique characteristics of existing features, and the new features generated using PCA do not represent the dataset well.

Some hybrid methods combine misuse and anomaly detection to mitigate the drawbacks of pattern-matching and their false positives [9], [10], [11]. However, these methods induce long detection times because of the combination of techniques, and they have difficulty detecting attack patterns hidden within the normal distribution [12]. Unknown attacks with normal patterns have recognizable features distributed at the decision boundaries of the trained model. Hence, newer methods have focused on detecting anomalies at the decision boundary.

An unknown attack refers to an attack that is not precollected or that the detection model has not learned, and a zero-day attack is a typical unknown attack. In addition, attacks that exhibit different behavior patterns from previously collected attacks are considered unknown attacks: they could be variants or are similar to normal. In cybersecurity, anomalies are primarily performed using anomaly detection. However, because anomaly detection determines all attacks as outliers, it is not clear whether a detected attack is a known type. A hybrid method is applied to detect unknown attacks by separating detected attacks into known and unknown types. Other research is generated new attacks through known attacks and using new attack for detecting unknown. Performance assessments deliberately remove one attack class from the train data and include the corresponding attack class in the test data to determine if an unlearned attack class is detected.

This study follows suit and proposes an improved method that applies discretization-based data preprocessing and decision-boundary data-point analysis to identify attacks similar to normal, resulting in the following contributions:

- An effectively trained anomaly detection model that accurately and efficiently classifies them into types and applies data discretization suitable for each.
- Reduced false positives because of decision boundary data point analysis.
- More accurate unknown attack detection based on entropy properties.

The remainder of this paper proceeds as follows. Section 2 introduces related works. Section 3 describes the research method and structure. Section 4 describes our experiment, and Section 5 summarizes the results and presents directions for future research.

II. RELATED WORKS

The learning models used for unknown attack detection have been enhanced over time using data preprocessing, ML, and various anomaly detection schemes, eventually leading to the methods employed in this study.

A. DATA DISCRETIZATION

Early on, researchers classified the knowledge discovery dataset (KDD) into three types and created an analysis method suitable for characterizing each [13], [14], whereas Aggarwal et al. classified features and analyzed detection performance [15]. Hashem et al. classified features for discretization [16], and another used discretization preprocessing to detect network intrusions [17] by classifying features via data analysis followed by type preprocessing. Discretization using equal-width binning to generate feature sections of specific sizes was applied in support of anomaly detection [18], and several discretization methods were applied to evaluate IDS performance [19]. Other studies focused on anomaly detection using entropy-based discretization [20], [21]. To effectively learn the data and ensure that each feature has a positive effect on model performance, a suitable discretization method was applied through feature analysis.

B. ANOMALY DETECTION

Most anomaly detection methods aim to identify data outside the range of the trained model (outliers) but do not determine their classes. A synthetic minority oversampling technique was used to generate attack data and a corresponding model using long short-term memory with an attention mechanism to detect unknown attacks [9]. DeepARMOUR was later proposed to detect tampered or corrupted data [10]. This technique removes meaningless features through a reduction process and uses random forest and graph learning models with multilayer perceptrons to provide classification results through voting. A novel hybrid model was proposed that uses decision trees (DTs) to identify known attacks and a support vector machine (SVM) to determine unknown ones [11]. AlEroud et al. generated profiles for known attacks and matched detections for classification purposes [22], whereas Bitaab et al. classified attacks using a DT while applying a Gaussian mixture model to detect normal or normal-like attacks [23]. Kamarudin et al. was proposed that detects known and unknown attacks using the LogitBoost algorithm [24]. In [25], a model is proposed that detects new attacks by extracting class features from learned data using a semi-supervised method to feed features to an SVM. This model learns to measure the similarity between features for accurate detection. Liu et al. classifies known and unknown attacks using a generative adversarial network (GAN) [26], which generates data and performs k-means clustering. Data exceeding a specific threshold in the

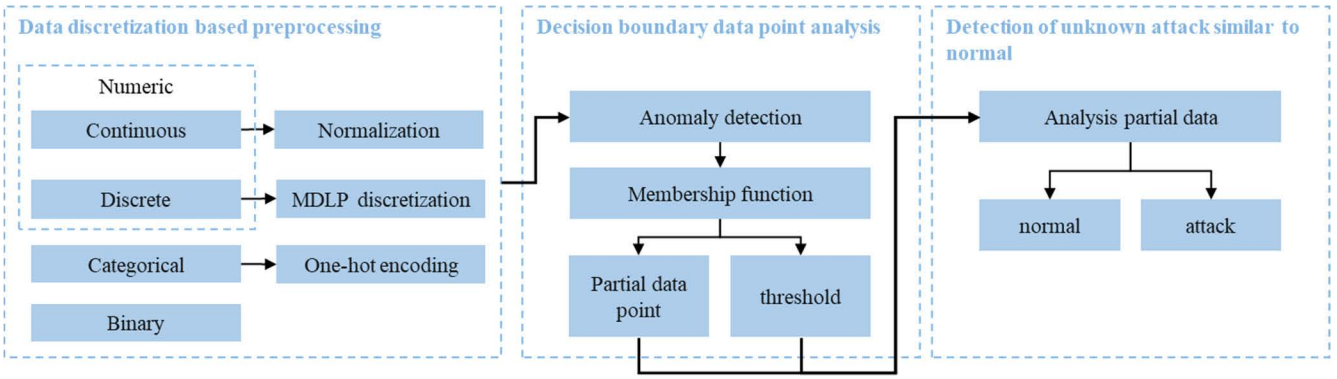

FIGURE 1. Framework for detecting unknown attacks that appear similar to normal network data patterns using data discretization and entropy. MDLP = Minimum description-length principle.

cluster is judged to be unknown. MalGAN, which generates adversarial examples based on malicious code using a GAN, was soon released [27]. The generated examples are similar to existing attacks and are classified as unknown attacks of different types. A method was then proposed to improve MalGAN's feature recognition capability using samples of malicious code, successfully detecting unknown attacks in experiments [28]. Chauhan et al. attempted to enable effective detection by training a GAN to create denial-of-service (DoS) attacks [29]. Lin et al. then proposed IDS GAN, which generates malicious traffic that evades detection to attack IDSs, improving their attacks through adversarial learning[30]. Zhang et al. proposed an open convolutional neural network (CNN) model by applying an OpenMax layer to calculate the probability of each class and locate the one with the highest probability [31]. A new Open-CNN model was soon released, which detects unknown attacks in drone networks [32], showing improved zero-day performance. Cruz et al. detected unknown attacks using a Weibull-calibrated SVM [33].

III. PROPOSED METHOD

This study attempts to achieve high detection performance and accurate data identification near the decision boundary by determining the characteristic loss of each feature when applying batch preprocessing and identifying the inaccuracy of distance similarities in high-dimensional data by measuring their distance-based similarities alongside the aforementioned problems of anomaly detection [34]. The method of detecting unknown attacks is illustrated in Fig. 1. First, our feature engineering process, unlike conventional batch preprocessing, classifies the datasets into three feature types and sets a discretization method suitable for each. During the decision boundary data point analysis, data located near the decision boundary of the model are identified using anomaly detection and fuzzy c-means (FCM), and we attempt to more accurately detect unknown attacks that appear normal in terms of their statistical network features by applying entropy to the partial data.

A. DATA DISCRETIZATION-BASED PREPROCESSING

In the feature engineering step, features are analyzed and divided into three types. The numeric type contains data in the form of numbers and can be further categorized into continuous and continuous within the interval types. Continuous data take the form of continuous numbers, whereas continuous within the interval data take the form of continuous numbers but they have a specific interval. Categorical types are divided into categories, and binary types comprise zeros and ones.

In our proposed method, to effectively learn the features of the three types, a preprocessing method suitable for each type is applied. For the numeric type, normalization is used for continuous data, and minimum description-length principle (MDLP) discretization is used for continuous within the interval data, whereas label encoding is used for the categorical type. Because the binary type consists of ones and zeroes, no other preprocessing is required.

B. DECISION-BOUNDARY DATA-POINT ANALYSIS

In this step, the data distributed near the decision boundary are identified using anomaly detection and FCM. iForest [35], the one-class SVM (OCSVM) [36], covariance [37], and local outlier factor (LOF) [38] methods are used for anomaly detection. FCM is applied based on the anomaly detection results. Unlike conventional clustering, this method measures the membership degree contained in each cluster set. For example, if we assume there are two clusters, c_i, then the membership degree of data x to c_1 and c_2 can be expressed as 0.3 for c_1 and 0.7 for c_2. This FCM characteristic means that data may belong to two or more clusters, unlike other methods that use binary values [39]. The threshold is obtained by applying the calculated membership degree to (1) based on the accuracy and false alarm rate. Points with high accuracy and low false alarm rate are used to select the optimal threshold, which is then used to identify data near the decision boundary.

$$\text{Theshold} = |\text{MD}(C_{normal}, x_i) - \text{MD}(C_{attacl}, x_i)|, \quad (1)$$

where MD signifies the membership degree of each cluster obtained by FCM, x_i is the total data, $i = 1, \cdots, n$,

TABLE 1. Detailed attack configuration by type.

Attack types	Detailed attack types
DoS	back, land, Neptune, pod, smurf, teardrop
Probe	ipseep, nmap, portsweep, satan
R2L	ftp_write, guess_passwd, imap, multihop, hpf, spy, warezclient, warezmaster
U2R	Buffer_overflow, loadmodule, perl, rootkit

Note: DoS = Denial of service; R2L = Remote-to-local; U2R = User-to-root.

TABLE 2. Normal and attack data ratio in NSL-KDD.

	Normal (%)	Attack (%)
Training data	53	47
Testing data	52	48

and C_{normal} and C_{attack} indicate normal and attack clusters, respectively.

C. DETECTION OF UNKNOWN ATTACKS SIMILAR TO NORMAL

To detect an unknown attack that looks like normal traffic, the entropy between normal and attack data near the decision boundary is calculated. This entropy is then used to reclassify the partial data into a class with lower information impurity between normal and attack, through which the model provides the detection result.

IV. EXPERIMENT

To verify the efficacy and performance of the proposed method, we conducted an experiment comprising two steps: 1) identify the attack data at the decision boundary using data point analysis, and 2) detect unknown attacks that resemble normal patterns.

A. EXPERIMENT DATA

KDD99 is the first benchmark dataset for intrusion detection, but it has some problems. Thus, NSL-KDD was used [40], [41] to address the issues of meaningless and duplicate data in previous works [42]. There were 41 features and 23 attack types, which were divided into four attack classes (Table 1).

As presented in Table 2, the training and test data comprised nearly identical proportions of normal and attack types. Although one may conclude that sufficient attack data were collected based on these items alone, as shown in Table 3, DoS accounted for most attacks, and there was substantially less data for user-to-root (U2R) and remote-to-local (R2L) than DoS and Probe.

B. EVALUATION METRICS

A confusion matrix was used to evaluate the performance of the proposed model. Accuracy, precision, recall, and F1-score measures are added to the matrix.

- Accuracy refers to ''normal'' vs. ''attack'' classification accuracy.
- Precision is the ratio of total true positives divided by the total number of true and false positives.
- Recall is the ratio of total relevant results correctly classified as true positives divided by the total number of true positives and false negatives.
- F-measure reflects the harmonic mean of precision and recall.

TABLE 3. Normal and attack types data ratio in NSL-KDD.

	Train data (%)	Test data (%)
Normal	53.45	51.66
DoS	36.45	30.54
Probe	9.25	5.88
U2R	0.04	0.19
R2L	0.78	11.70

TABLE 4. Classification and preprocessing methods according to feature type.

Type		Feature	Method
Numeric	*Continuous*	'duration', 'src_bytes', 'dst_bytes', 'hot', 'num_failed_logins', 'num_compromised', 'num_root', 'num_file_creations', 'num_shells', 'num_access_files' 'wrong_fragment', 'urgent', 'su_attempted', 'count', 'srv_count', 'serror_rate', 'srv_serror_rate', 'rerror_rate', 'srv_rerror_rate', 'same_srv_rate', 'diff_srv_rate', 'srv_diff_host_rate',	Normalization
	Continuous within the interval	'dst_host_count', 'dst_host_srv_count', 'dst_host_same_srv_rate', 'dst_host_diff_srv_rate', 'dst_host_same_src_port_rate', 'dst_host_srv_diff_host_rate', 'dst_host_serror_rate', 'dst_host_srv_serror_rate', 'dst_host_rerror_rate', 'dst_host_srv_rerror_rate'	MDLP discretization
Categorical		'protocol_type', 'service', 'flag'	Label encoding
Binary		'land', 'logged_in', 'root_shell', 'num_outbound_cmds', 'is_host_login', 'is_guest_login'	-

C. DATA DISCRETIZATION-BASED PREPROCESSING

Features were classified into three types, and a data discretization method suitable for each type was applied. Table 4 presents the NSL-KDD feature classification according to each feature type.

The numeric type contains 32 features, the categorical type contains three, and the binary type contains six. Min–max normalization was used for the continuous type, and MDLP discretization was used for the continuous within the interval type [43]. Categorical data were converted into a form suitable for model training using label encoding with no additional preprocessing.

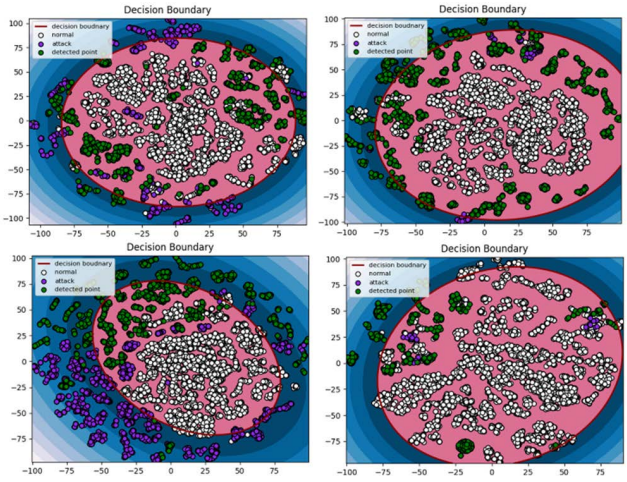

FIGURE 2. Decision boundary (red line) of covariance-based anomaly detection model and distribution of normal (white dots), attack (purple dots), and data near identified decision boundary (green dots). Clockwise from top-right: Distribution of R2L, probe, dos, and U2R.

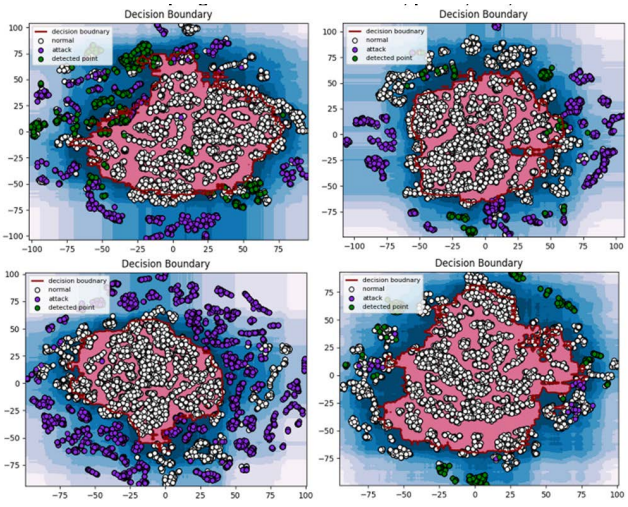

FIGURE 3. Decision boundary (red line) of iForest-based anomaly detection model and distribution of normal (white dots), attack (purple dots), and data near identified decision boundary (green dots). Clockwise from top-right: Distribution of R2L, probe, dos, and U2R.

D. IDENTIFICATION OF ATTACK DATA THROUGH DECISION-BOUNDARY DATA-POINT ANALYSIS

Anomaly detection and FCM were used to identify data near the decision boundary and calculate the corresponding thresholds. Figs. 2 and 3 show the decision boundary of the anomaly detection model alongside the normal data, attack data, and data near the decision boundary using T-distributed stochastic neighbor embedding (T-SNE) [44]. As shown, the data (green dots) identified near the edge of the decision boundary (red line) of the anomaly detection model are distributed, indicating that the data located near the decision boundary can be identified.

E. DETECTING UNKNOWN ATTACKS SIMILAR TO NORMAL

Using the identified data near the decision boundary, we verified the classification results using measurements of entropy with normal and attack types. Tables 5–8 compare detection performances for unknown attacks that appear normal via the identification and classification of data near the decision boundary compared to those using iForest, OCSVM, covariance, and LOF. Cross-validation was applied to the evaluation, prior to application to the algorithm, and parameter optimization for each algorithm was performed, as shown in the Table 9. We confirmed that the overall performance improved when applying our proposed method; the lower the anomaly detection performance prior to applying our method, the larger the increase. This can be attributed to the fact that performance increases with lesser false positive data near the decision boundary.

TABLE 5. Comparison verification results between forest-based anomaly detection and proposed method.

		accuracy	precision	recall	f1-score	false alarm
iForest	DoS	**0.9052**	**0.8779**	**0.9668**	**0.9202**	**0.1751**
	Probe	0.9638	0.9877	0.9668	0.9771	0.0483
	R2L	0.7872	0.7993	0.9668	0.8751	0.8173
	U2R	0.9634	0.9962	0.9668	0.9813	0.5373
Proposed method	DoS	**0.9052**	**0.8779**	**0.9668**	**0.9202**	**0.1751**
	Probe	**0.9674**	**0.9917**	**0.9675**	**0.9795**	**0.0326**
	R2L	**0.7967**	**0.8114**	**0.9593**	**0.8792**	**0.7508**
	U2R	**0.9722**	**0.9967**	**0.9752**	**0.9858**	**0.4627**

TABLE 6. Comparison verification results between OCSVM-based anomaly detection and proposed method.

		accuracy	precision	recall	f1-score	false alarm
OCSVM	DoS	0.7795	0.7423	0.9345	0.8274	0.4224
	Probe	0.8649	0.9905	0.9345	0.9172	0.4143
	R2L	0.7510	0.7840	0.9345	0.8527	0.8666
	U2R	0.9292	0.9939	0.9345	0.9633	0.8358
Proposed method	DoS	**0.8457**	**0.8168**	**0.9373**	**0.8729**	**0.2736**
	Probe	**0.9379**	**0.9666**	**0.9554**	**0.9610**	**0.1326**
	R2L	**0.7940**	**0.8215**	**0.9363**	**0.8752**	**0.6849**
	U2R	**0.9305**	**0.9964**	**0.9334**	**0.9639**	**0.4925**

TABLE 7. Comparison verification results between covariance-based anomaly detection and proposed method.

		accuracy	precision	recall	f1-score	false alarm
Covariance	DoS	0.8094	0.7972	0.8892	0.8407	0.2945
	Probe	0.8581	0.9304	0.8892	0.9093	0.2668
	R2L	0.7691	0.8249	0.8892	0.8558	0.6354
	U2R	0.8872	0.9969	0.8892	0.9400	0.4030
Proposed method	DoS	**0.8702**	**0.8294**	**0.9700**	**0.8942**	**0.2598**
	Probe	**0.9546**	**0.9799**	**0.9630**	**0.9714**	**0.0793**
	R2L	**0.8090**	**0.8449**	**0.9214**	**0.8815**	**0.5695**
	U2R	**0.9575**	**0.9976**	**0.9594**	**0.9781**	**0.3284**

TABLE 8. Comparison verification results between LOF-based anomaly detection and proposed method.

		accuracy	precision	recall	f1-score	false alarm
LOF	DoS	0.7484	0.9176	0.6098	0.7327	0.0713
	Probe	0.6775	0.9795	0.6098	0.7517	0.0512
	R2L	0.6267	0.8663	0.6099	0.7158	0.3168
	U2R	0.6081	0.9926	0.6099	0.7556	0.6567
Proposed method	DoS	**0.7928**	**0.9185**	**0.6954**	**0.7915**	**0.0803**
	Probe	**0.7565**	**0.9777**	**0.7121**	**0.824**	**0.0653**
	R2L	**0.6951**	**0.8876**	**0.6922**	**0.7778**	**0.2950**
	U2R	**0.6968**	**0.9952**	**0.6981**	**0.8206**	**0.4925**

TABLE 9. Optimizing algorithm parameters.

Algorithm	Parameter
iForest	**bootstrap: false, estimators: 50**
OCSVM	**kernel: linear, nu: 0.1, tol: 0.001**
Covariance	**support fraction: 0.1**
LOF	**novelty: true**

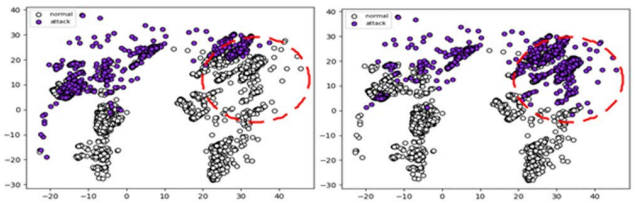

FIGURE 4. Re-identification results of unknown probe attacks that appear normal through the proposed method (right: before identification, left: after identification).

There was no significant difference in performance with iForest, but the false alarm rate decreased, signifying that unknown attacks that appeared normal were accurately detected. Meanwhile, the proposed method outperformed the others by identifying data near the decision boundary and reclassifying unknown attacks that appeared normal. Accuracy improved by 5–7% for DoS attacks, 7–10% for probes, 4–7% for R2L, and 1–9% for U2R, and the false alarm rate decreased overall.

Figs. 4 and 5 visualize the detection results before and after applying the proposed method to confirm the unknown attacks. The red line indicates the unknown attacks at the decision boundary, and the results show that they were successfully detected by applying the proposed method.

We compared the proposed method to successful methods from previous studies (Table 10-12). In True Positive Rate(TPR), it was confirmed that U2R and R2L showed better performance than previous studies, and we confirmed Probe and R2L have high accuracy. Through this comparative verification, we confirmed that the proposed method can identify attacks that appear normal and is comparable to or better than existing methods.

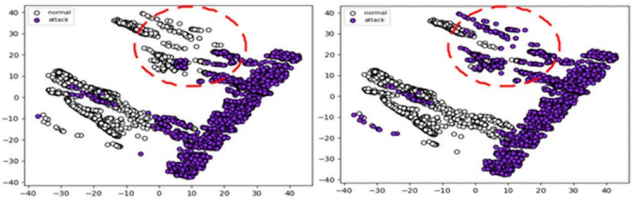

FIGURE 5. Re-identification results of unknown denial-of-service attacks that appear normal through the proposed method (right: before identification, left: after identification).

TABLE 10. Comparison verification results of true positive rate performance with previous studies.

Model	DoS	Probe	U2R	R2L
[45]	0.9034	0.9153	0.5400	0.8243
[46]	**0.9996**	**0.9977**	0.8082	0.9734
Proposed Method	0.9668	0.9675	**0.9593**	**0.9752**

TABLE 11. Comparison verification results of accuracy performance with previous studies.

Model	ACCURACY
[34]	0.8098
[35]	0.9049
[45]	0.9136
[46]	**0.9989**
Proposed Method	0.9156

TABLE 12. Comparison verification results of detail attack type accuracy performance with previous studies.

Model	DoS	Probe	U2R	R2L
[24]	**0.9982**	0.5483	N/A	0.1667
[34]	0.8137	0.8793	**0.8288**	0.7176
[35]	0.9052	0.9638	0.7872	0.9634
Proposed Method	0.9052	**0.9674**	0.7967	**0.9722**

V. CONCLUSION AND FUTURE WORK

This study proposed a method for detecting unknown cyber-attacks with behavior patterns similar to normal patterns based on data discretization at the decision boundary, data point-by-data point. Hence, unknown attacks were identified by their discrete features using the NSL-KDD for IDS to apply preprocessing for every feature type. Subsequently, we improved detection performance by reclassifying attacks based on entropy characterizations.

Using NSL-KDD, the proposed method was evaluated in terms of accuracy, precision, recall, F1-score, and false alarm rate. In the case of four or more anomaly detection algorithms, models suitable for data were constructed through

hyperparameter optimization. We visualized the data using T-SNE and verified that the data items near the decision boundary had been properly identified. Moreover, visualizations of the detection results before and after applying the proposed method confirmed that unknown attacks were detected. The results confirmed that the false positive rate was reduced. Finally, our method's performance was verified through comparisons with results achieved in prior studies.

In a future study, we plan to evaluate our model on other network datasets to test its scalability and mitigate the class imbalance problems that continue to pervade IDS learning models. In this study, our high-performance anomaly detection model failed to accurately identify data near the decision boundary occasionally. Therefore, we also plan to develop improved methods to overcome this phenomenon.

REFERENCES

[1] P. Duessel, C. Gehl, U. Flegel, S. Dietrich, and M. Meier, "Detecting zero-day attacks using context-aware anomaly detection at the application-layer," *Int. J. Informat. Secur.*, vol. 16, no. 5, pp. 475–490, 2017.

[2] X. Sun, J. Dai, P. Liu, A. Singhal, and J. Yen, "Using Bayesian networks for probabilistic identification of zero-day attack paths," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 10, pp. 2506–2521, Oct. 2018.

[3] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *J. Netw. Comput. Appl.*, vol. 60, pp. 19–31, Jan. 2016.

[4] M. Ahmed, A. N. Mahmood, and J. Hu, "Outlier detection," in *The State of the Art in Intrusion Prevention and Detection*, vol. 44, A. S. K. Pathan, Ed. Boca Raton, FL, USA: CRC Press, 2014, ch. 1, pp. 3–21.

[5] J. Zhao, S. Shetty, J. W. Pan, C. Kamhoua, and K. Kwiat, "Transfer learning for detecting unknown network attacks," *EURASIP J. Informat. Secur.*, vol. 2019, no. 1, pp. 1–13, 2019.

[6] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection: A review," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–38, Mar. 2021.

[7] S. Lakhina, S. Joseph, and B. Verma, "Feature reduction using principal component analysis for effective anomaly-based intrusion detection on NSL-KDD," *Int. J. Eng. Sci. Technol.*, vol. 2, no. 6, pp. 1790–1799, 2010.

[8] H. Benaddi, K. Ibrahimi, and A. Benslimane, "Improving the intrusion detection system for NSL-KDD dataset based on PCA-fuzzy clustering-KNN," in *Proc. 6th Int. Conf. WINCOM*, 2018, pp. 1–6.

[9] P. Lin, K. Ye, and C. A. Xu, "Dynamic network anomaly detection system by using deep learning techniques," in *Proc. Int. Conf. Cloud Comput.*, 2019, pp. 161–176.

[10] Y. Ji, B. Bowman, and H. H. Huang, "Securing malware cognitive systems against adversarial attacks," in *Proc. IEEE ICCC*, Jul. 2019, pp. 1–9.

[11] G. Kim, S. Lee, and S. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection," *Expert Syst. Appl.*, vol. 41, no. 4, pp. 1690–1700, Mar. 2014.

[12] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "FFSc: A novel measure for low-rate and high-rate DDoS attack detection using multivariate data analysis," *Secur. Commun. Netw.*, vol. 9, no. 13, pp. 2032–2041, 2016.

[13] H. S. Chae, B. O. Jo, S. H. Choi, and T. K. Park, "Feature selection for intrusion detection using NSL-KDD," *Recent Adv. Comput. Sci.*, vol. 20132, pp. 184–187, Nov. 2013.

[14] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl.*, Jul. 2009, pp. 1–6.

[15] P. Aggarwal and S. K. Sharma, "Analysis of KDD dataset attributes-class wise for intrusion detection," *Proc. Comput. Sci.*, vol. 57, pp. 842–851, Jan. 2015.

[16] S. Hashem and H. Adil, "Denial of service intrusion detection system (IDS) based on Naïve Bayes classifier using NSL KDD and KDD cup 99 datasets," *J. Al-Rafidain Univ. College Sci.*, vol. 2, no. 40, pp. 206–231, 2017.

[17] J. Yoo, B. Min, S. Kim, D. Shin, and D. Shin, "Study on network intrusion detection method using discrete pre-processing method and convolution neural network," *IEEE Access*, vol. 9, pp. 142348–142361, 2021.

[18] A. S. A. Aziz, A. T. Azar, A. E. Hassanien, and S. E.-O. Hanafy, "Continuous features discretization for anomaly intrusion detectors generation," in *Soft Computing in Industrial Applications*. Cham, Switzerland: Springer, 2014, pp. 209–221.

[19] V. Bolon-Canedo, N. Sanchez-Marono, and A. Alonso-Betanzos, "A combination of discretization and filter methods for improving classification performance in KDD cup 99 dataset," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2009, pp. 359–366.

[20] R. Ratti, S. R. Singh, and S. Nandi, "Towards implementing fast and scalable network intrusion detection system using entropy based discretization technique," in *Proc. 11th ICCCNT*, 2020, pp. 1–7.

[21] H. F. Eid, A. T. Azar, and A. E. Hassanien, "Improved real-time discretize network intrusion detection system," in *Proc. 7th Int. Conf. BIC-TA*, 2013, pp. 99–109.

[22] A. AlEroud and G. Karabatis, "A contextual anomaly detection approach to discover zero-day attacks," in *Proc. Int. Conf. Cyber Secur.*, Dec. 2012, pp. 40–45.

[23] M. Bitaab and S. Hashemi, "Hybrid intrusion detection: Combining decision tree and Gaussian mixture model," in *Proc. 14th ISCISC*, 2017, pp. 8–12.

[24] M. H. Kamarudin, C. Maple, T. Watson, and N. S. Safa, "A logitboost-based algorithm for detecting known and unknown web attacks," *IEEE Access*, vol. 5, pp. 26190–26200, 2017.

[25] S. Huda, S. Miah, M. M. Hassan, R. Islam, J. Yearwood, M. Alrubaian, and A. Almogren, "Defending unknown attacks on cyber-physical systems by semi-supervised approach and available unlabeled data," *Inf. Sci.*, vol. 379, pp. 211–228, Feb. 2017.

[26] A. Liu, Y. Wang, and T. Li, "SFE-GACN: A novel unknown attack detection under insufficient data via intra categories generation in embedding space," *Comput. Secur.*, vol. 105, Jun. 2021, Art. no. 102262.

[27] W. Hu and Y. Tan, "Generating adversarial malware examples for black-box attacks based on GAN," 2017, *arXiv:1702.05983*.

[28] M. Kawai, K. Ota, and M. Dong, "Improved MalGAN: Avoiding malware detector by learning cleanware features," in *Proc. ICAIIC*, 2019, pp. 40–45.

[29] R. Chauhan and S. S. Heydari, "Polymorphic adversarial DDoS attack on IDS using GAN," in *Proc. IEEE ISNCC*, Oct. 2020, pp. 1–6.

[30] Z. Lin, Y. Shi, and Z. Xue, "IDSGAN: Generative adversarial networks for attack generation against intrusion detection," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*, 2022, pp. 79–91.

[31] Y. Zhang, J. Niu, D. Guo, Y. Teng, and X. Bao, "Unknown network attack detection based on open set recognition," *Proc. Comput. Sci.*, vol. 174, pp. 387–392, Jan. 2020.

[32] Z. Zhang, Y. Zhang, J. Niu, and D. Guo, "Unknown network attack detection based on open-set recognition and active learning in drone network," *Trans. Emerg. Telecomnmun. Technol.*, vol. 33, no. 10, Jan. 2021, Art. no. e4212.

[33] S. Cruz, C. Coleman, E. M. Rudd, and T. E. Boult, "Open set intrusion recognition for fine-grained attack categorization," in *Proc. IEEE Int. Symp. Technol. Homeland Secur. (HST)*, Apr. 2017, pp. 1–6.

[34] G.-Y. Shin, D.-W. Kim, S.-S. Kim, and M.-M. Han, "Unknown attack detection: Combining relabeling and hybrid intrusion detection," *Comput., Mater. Continua*, vol. 68, no. 3, pp. 3289–3303, 2021.

[35] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Mining*, 2008, pp. 413–422.

[36] B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, 2001.

[37] P. J. Rousseeuw and K. Van Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.

[38] M. M. Breunig, H. P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, 2000, pp. 93–104.

[39] D. J. Bora and D. A. K. Gupta, "A comparative study between fuzzy clustering algorithm and hard clustering algorithm," 2014, *arXiv:1404.6059*.
[40] *NSL-KDD Dataset*. Accessed: Oct. 2022. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html
[41] *KDD Cup 1999 Data*. Accessed: Oct. 2022. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
[42] A. Thakkar and R. Lohiya, "A review of the advancement in intrusion detection datasets," *Proc. Comput. Sci.*, vol. 167, pp. 636–645, Jan. 2020.
[43] U. Fayyad and K. Irani, "Multi-interval discretization of continuous-valued attributes for classification learning," in *Proc. IJCAI*, 1993, pp. 1022–1029.
[44] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.
[45] Y. Imrana, Y. Xiang, L. Ali, and Z. Abdul-Rauf, "A bidirectional LSTM deep learning approach for intrusion detection," *Expert Syst. Appl.*, vol. 185, Dec. 2021, Art. no. 115524.
[46] Y. Zhou, T. A. Mazzuchi, and S. Sarkani, "M-AdaBoost-A based ensemble system for network intrusion detection," *Expert Syst. Appl.*, vol. 162, Dec. 2020, Art. no. 113864.

GUN-YOON SHIN received the bachelor's degree in interactive media convergence and the master's degree in computer engineering from Gachon University, South Korea, in 2017 and 2018, respectively, where he is currently pursuing the Ph.D. degree with the Department of Computer Engineering. His research interests include authorship attribution, unknown attack detection, network anomaly detection, information security, machine learning, and artificial intelligence.

DONG-WOOK KIM received the bachelor's degree in computer software and the master's degree in computer engineering from Gachon University, South Korea, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree with the Department of Computer Engineering. His research interests include insider threats, information security, data mining, and machine learning.

MYUNG-MOOK HAN received the M.S. degree in computer science from the New York Institute of Technology, in 1987, and the Ph.D. degree in information engineering from Osaka City University, in 1997. From 2004 to 2005, he was a Visiting Professor at the Georgia Tech Information Security Center (GTISC), Georgia Institute of Technology. He is currently a Professor with the Department of Software, Gachon University, South Korea. His research interests include information security, intelligent systems, data mining, and big data.