KOSDI
ISSN : 1976-2208 (Print)
ISSN : 2671-5287 (On-line)
www.sodi.or.kr

Journal of the Society of Disaster Information
Vol. 19, No. 4, pp. 924-935, December 2023
https://doi.org/10.15683/kosdi.2023.12.31.924

Original Article

SHAP 기반 NSL-KDD 네트워크 공격 분류의 주요 변수 분석

Analyzing Key Variables in Network Attack Classification on NSL-KDD Dataset using SHAP

이상덕[1] · 김대규[2] · 김창수[3]*
Sang-duk Lee[1], Dae-gyu Kim[2], Chang Soo Kim[3]*

[1]Ph.D Candidate, Big Data Collaborative. 138, Central Police Academy, Suhoeri-ro, Suanbo-myeon, Chungju-si, Chungcheongbuk-do, Republic of Korea.
[2]Ph.D Candidate, Department of IT Convergence and Application Engineering, Pukyong National University, Busan, Republic of Korea
[3]Professor, Department of IT Convergence and Application Engineering, Pukyong National University, Busan, Republic of Korea

*Corresponding author: Chang Soo Kim, cskim@pknu.ac.kr

ABSTRACT

Purpose: The central aim of this study is to leverage machine learning techniques for the classification of Intrusion Detection System (IDS) data, with a specific focus on identifying the variables responsible for enhancing overall performance. **Method:** First, we classified ‘R2L(Remote to Local)’ and ‘U2R (User to Root)’ attacks in the NSL-KDD dataset, which are difficult to detect due to class imbalance, using seven machine learning models, including Logistic Regression (LR) and K-Nearest Neighbor (KNN). Next, we use the SHapley Additive exPlanation (SHAP) for two classification models that showed high performance, Random Forest (RF) and Light Gradient-Boosting Machine (LGBM), to check the importance of variables that affect classification for each model. **Result:** In the case of RF, the 'service' variable and in the case of LGBM, the 'dst_host_srv_count' variable were confirmed to be the most important variables. These pivotal variables serve as key factors capable of enhancing performance in the context of classification for each respective model. **Conclusion:** In conclusion, this paper successfully identifies the optimal models, RF and LGBM, for classifying 'R2L' and 'U2R' attacks, while elucidating the crucial variables associated with each selected model.

Keywords: NSL-KDD, Remote to Local (R2L), User to Root (U2R), eXplainalble Artificial Intelligence (XAI), SHapley Additive exPlanation (SHAP)

Received | 14 November, 2023
Revised | 12 December, 2023
Accepted | 14 December, 2023

Introduction

With the widespread adoption of cloud services and IoT devices, there has been a dramatic surge in network traffic, highlighting the increasing importance of network communication security. If the level of network communication security is low and personal information leakage occurs due to hacking, personal personal information is at stake and has a great ripple effect, so this can be seen as a social disaster (So et al., 2021). In anticipation of these security threats, extensive research is underway in the field of large-scale network traffic analysis and intrusion detection utilizing IDS. IDS represents a real-time network resource monitoring system designed to detect and promptly notify users of any anomalous behavior, with corresponding response capabilities. In the past, attacks were primarily detected using patterns (Bace et al., 2001). But with the advancement of hardware capable of handling large volumes of data, researchers are now focusing on AI-based detection techniques, leading to the development of high-performance methods that are gaining recognition (Khraisat et al., 2019; Jeong et al., 2020). Nevertheless, AI-based detection techniques have the limitation of being unable to explain the specific variables that played a significant role, as they merely present performance results for each model in a black box manner. In this paper, our aim is to examine the key variables within the network attack classification model through the application of the SHapley Additive exPlanation (SHAP) methodology using eXplainable Artificial Intelligence (XAI) techniques, which can address the limitations associated with the aforementioned detection methods. The structure of the paper is as follows. We begin our study by reviewing prior research concerning the NSL-KDD dataset and SHAP. Following this, we conduct experiments, analyze the results using the NSL-KDD dataset and SHAP, and conclude by presenting our findings.

Datasets and Related Research

NSL-KDD Dataset

In this paper, we conducted a study using NSL-KDD (http://www.unb.ca/cic/research/datasets/nsl.html), a dataset widely used in IDS analysis. The Knowledge Discovery in Database (KDD) CUP 99 (http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html) dataset, which was extensively employed prior to the NSL-KDD dataset, posed limitations for machine learning applications, primarily stemming from issues such as imbalanced data distribution and a high volume of duplicate records. To solve this problem, NSL-KDD was proposed. There are five classes in the NSL-KDD dataset: 'Normal', 'Denial of Service (DoS)', 'Probe', 'Remote to Local (R2L)', and 'User to Root (U2R)'. The training dataset consists of 21 attack types, the test dataset consists of 22 attack types, and each record has 41 features. In this paper, because there is a part of adjusting the training data, the existing test data 'KDDTest+.txt' is not used, and 'KDDTrain+.txt'

Table 1. Composition of NSL-KDD Train dataset 'KDDTrain+.txt'

Dataset	Total	Normal	Dos	Probe	R2L	U2R
KDDTrain+	125,973	67,343	45,927	11,656	995	52

Table 2. Description of NSL-KDD Train dataset 'KDDTrain+.txt'

Attack type	Descriptions
DoS	Attacks that disrupt normal access to network services
Probe	An attack that scans the attack target and collects target information
R2L	An attack that illegally elevates normal user privileges to administrator privileges
U2R	Attacks that illegally access the local area from an external network

is used for both training and test data. Table 1, 2 shows the composition and description of the 'KDDTrain+.txt' dataset. In this paper, the existing attack type labels were changed and configured as shown in Table 3.

Table 3. Attack type and counts of NSL-KDD Train dataset

Existing	Change (amount)
'Normal'	'Normal' (67,342)
'apache2', 'back', 'land', 'neptune', 'mailbomb', 'pod', 'processtable', 'smurf', 'teardrop', 'udpstorm', 'worm'	'DoS' (45,927)
'ipsweep', 'mscan', 'nmap', 'portsweep', 'saint', 'satan'	'Probe' (11,656)
'ftp_write', 'guess_passwd', 'httptunnel', 'imap', 'multihop', 'named', 'phf', 'sendmail', 'snmpgetattack', 'snmpguess', 'spy', 'warezclient', 'warezmaster', 'xlock', 'xsnoop'	'R2L' (995)
'buffer_overflow', 'loadmodule', 'perl', 'ps', 'rootkit', 'sqlattack', 'xterm'	'U2R' (52)

XAI(eXplainable Artificail Intelligence) - SHAP

Numerous methodologies exist for XAI techniques, including Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016) and SHAP (Shapley, 1953). In this paper, we use SHAP, a model-agnostic technique. LIME assigns weights by assessing the similarity between an instance and the original one, while SHAP measures weights for sampled instances based on values obtained from Shapley value estimation. SHAP measures the contribution of each feature in model predictions. This is a method of calculating and interpreting the Sharply value of each player in the game using coalitional game theory. Recently, numerous studies have applied the SHAP technique to diverse IDS datasets. M. Wang et al.(2020) introduced a SHAP-based framework to enhance the interpretability of IDS, utilizing the NSL-KDD dataset to provide both global and local descriptions. In this paper, the analysis results generated from the proposed framework are consistent with the characteristics of specific attacks and are said to help cybersecurity experts make better decisions when designing IDS and identify the characteristics of various attacks. Wang et al.(2021) said that by classifying the CICIDS2017 dataset using LightGBM and CNN models and providing global and local explanations using SHAP, the analysis results can be interpreted in several directions. It is said that this allows understanding the characteristics of various network attacks and helps improve performance and optimize the system. Le et al.(2022) conducted a study to explain the improvement of IDS attack detection performance and machine learning model prediction using IoT-based IDS datasets NF-BoT-IoT-v2, NF-ToN-IoT-v2, and IoTDS20. In the paper, an ensemble tree machine learning model containing two models, Decision Tree (DT) and Random Forest (RF), was explained based on SHAP to improve the attack type detection rate. Wali et al.(2021) conducted a study using the CSE-CIC IDS 2018 dataset. In the paper, a new IDS that combines the RF algorithm and SHAP is presented, and after interpreting the prediction results through the

SHAP framework, an IDS that can identify all types of malicious content in network traffic is presented. These IDSs facilitate a transparent decision-making approach by evaluating model descriptions developed during the development and evaluation phases to increase user trust and maintain operational integrity.

In this paper, we use SHAP to find a model that appropriately classifies the 'R2L' and 'U2R' classes that are unbalanced in NSL-KDD, and to analyze what variables are important in classification. To this end, after going through the steps of detecting 'DoS' and 'Probe' attack types and classifying 'R2L' and 'U2R' attack types using seven classification models such as 'Logistic Regression' and 'K-Nearest', the optimal Select two classification models and find out which variable contributed the most based on SHAP. As a result, the 'service' variable in the case of RF and the 'dst_host_srv_count' variable in the case of LGBM were confirmed to be the most important variables. It can be seen that the important variables identified in this way are variables that can improve performance when classifying using each model.

Therefore, in this paper, we were able to identify RF and LGBM, which showed high performance among the models that classify 'R2L' and 'U2R' attacks, and the important variables for each model.

Experiment and Results

Experimental Dataset

The biggest problem with the NSL-KDD dataset is that the classes in the dataset are extremely imbalanced. The smallest class of 'U2R' is 52 in the training dataset, which is 0.001% of the 45,927 DoS, the most common attack type. If you use the training data as is, ignoring the imbalance of these classes, the amount of learning for the 'DoS' class in the model will increase. Therefore, there may be a bias toward 'R2L' and 'U2R', which have a relatively small amount of data, and although the classification of the 'DoS' class may be accurate in the actually created model, the classification of 'R2L' and 'U2R' is not made. Results may appear. :in the actual model, the classification of the 'DoS' class may be correct, but the classification of 'R2L' and 'U2R' may not be correct. Fig. 1 is the experimental design flow chart of this paper.

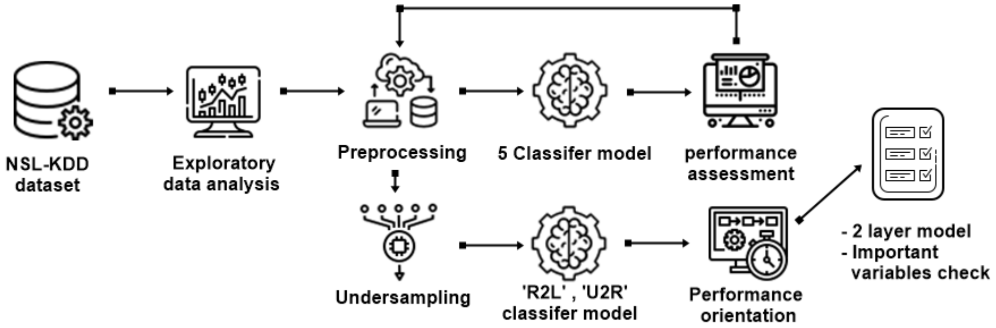

Fig. 1. Experimental design flow chart

Fig. 2 is a confusion matrix showing the prediction results of building a RF classification model using the NSL-KDD dataset. The built model shows an overall accuracy of 0.998, but 'R2L' shows an accuracy of 0.9597. In particular, 'U2R' accurately predicted only 0.5384. Even in models that show such high performance, the accuracy of classification for 'R2L' and 'U2R' is significantly reduced due to data imbalance, and in fact, detection of the 'U2R' class is particularly difficult in models built in this way.

To solve this problem, we used an under-sampling method that reduces a large number of data to a small number of data, which we tried to overcome by using the Changing the dataset by under sampling may result in information loss due to data removal, but it will increase the learning opportunity for 'R2L' and 'U2R', which have never been trained due to their relatively small number compared to other datasets, to create a classification model that can classify these attacks.

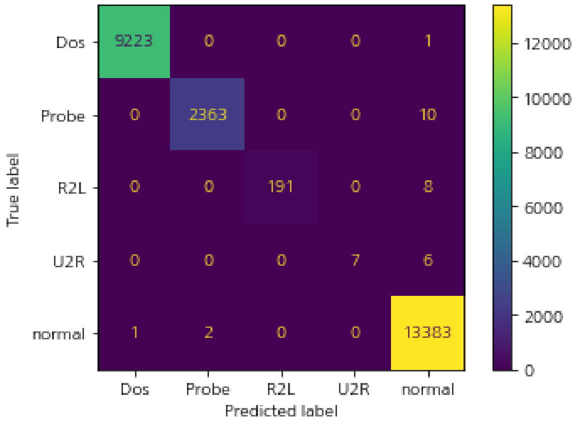

Fig. 2. Confusion matrix of classification results from a random forest model

Preprocessing

In order to accurately classify and analyze key variables for the 'R2L' and 'U2R' attack types, which are the targets of this study, we excluded other classes and performed undersampling as shown in Table 4 to reduce the number of datasets and further ' We created a dataset that can classify only the 'R2L' and 'U2R' classes.

Table 4. Modified NSL-KDD dataset

Total	Normal	R2L	U2R
2,047	1,000	995	52

The NSL-KDD dataset consists of a total of 41 variables, including 6 binary variables, 32 continuous variables, and 3 categorical variables. Among these, 32 continuous variables were standardized using MinMax-scaler, and 3 categorical variables were converted to integer type using LabelEncoder. The variables used are shown in Table 5.

Table 5. NSL-KDD dataset features

No	Variable	Type	No	Variable	Type
1	Duration	Continuous	22	Is_guest_login	Binary
2	Protocol_type	Categorical	23	Count	Continuous
3	Service	Categorical	24	Srv_count	Continuous
4	Flag	Categorical	25	Serror_rate	Continuous
5	Src_bytes	Continuous	26	Srv_serror_rate	Continuous
6	Dst_bytes	Continuous	27	Rerror_rate	Continuous
7	Land	Binary	28	Srv_rerror_rate	Continuous
8	Wrong_fragment	Continuous	29	Same_srv_rate	Continuous
9	Urgent	Continuous	30	Diff_srv_rate	Continuous
10	Hot	Continuous	31	Srv_diff_host_rate	Continuous
11	Num_failed_logins	Continuous	32	Dst_host_count	Continuous
12	Logged_in	Binary	33	Dst_host_srv_count	Continuous
13	Num_compromised	Continuous	34	Dst_host_same_srv_rate	Continuous
14	Root_shell	Binary	35	Dst_host_diff_srv_rate	Continuous
15	Su_attempt	Binary	36	Dst_host_same_src_port_rate	Continuous
16	Num_root	Continuous	37	Dst_host_srv_diff_host_rate	Continuous
17	Num_file_creations	Continuous	38	Dst_host_serror_rate	Continuous
18	Num_shells	Continuous	39	Dst_host_srv_serror_rate	Continuous
19	Num_access_files	Continuous	40	Dst_host_rerror_rate	Continuous
20	Num_outbound_cmds	Continuous	41	Dst_host_srv_rerror_rate	Continuous
21	Is_hot_login	Binary			

Create and Evaluate Classification Models

In this study, we created a pipeline for classification models using 'Logistic Regression (LR)', 'K-Nearest (KNN)', 'Support Vector Classifier (SVC)', 'Decision Tree (DT) Classifier', 'Random Forest (RF) Classifier', 'Light Gradient-Boosting Machine (LGBM) Classifier', 'Gradient Boosting Machine (GBM) Classifier', and after training, the models were The average accuracy of the cross validation results was evaluated. The evaluation results of the learned model are shown in Table 6.

Table 7 shows the results of parameter adjustment using the GridSearchCV technique for 'RF' and 'LGBM', which showed high accuracy among the above evaluation models.

As a result of the classification report of model, the accuracy score of Random Forest is 0.98. Even in the case of 'U2R', which has a lower result value compared to 'R2L' and 'Normal', the precision, which is the ratio of what is actually 'U2R' among what is classified as 'U2R', is 1.00, and it is confirmed that the classification of 'U2R' is performed correctly, but the recall, which is the ratio of what is actually classified as 'U2R', is 0.78. As a result, the F1-Score, which is the harmonic mean, is 0.84, and it is confirmed that Random Forest can classify 'U2R'.

Table 6. Measure the accuracy of each model

Model name	Accuracy
Logistic Regression	93.9500%
KNeighbors Classifier	95.7899%
SVC	95.8500%
Decision Tree Classifier	96.7600%
Random Forest Classifier	97.9200%
Gradient Boosting Classifier	97.8600%
LGBM Classifier	98.1700%

Table 7. Classification report of model

Model name	y	Accuracy	Precision	Recall	F1-Score
Random Forest Classifier	R2L	0.98	0.99	0.98	0.98
	U2R		1.00	0.73	0.84
	Normal		0.97	0.99	0.98
LGBM Classifier	R2L	0.98	0.99	0.99	0.99
	U2R		1.00	0.64	0.78
	Normal		0.97	0.99	0.98

In the case of LGBM, the accuracy score was found to be 0.98, which is the same as Random Forest, but the recall, which is the ratio of the results classified as 'U2R' to the actual results classified as 'U2R', was found to be 0.68, which is lower than Random Forest. As a result, the F1-Score, which is the harmonic mean, was also found to be 0.78, which is lower than Random Forest.

The results of confirming the confusion matrix for the classification values of the two models showing fairly high performance are shown in Fig. 3 and 4.

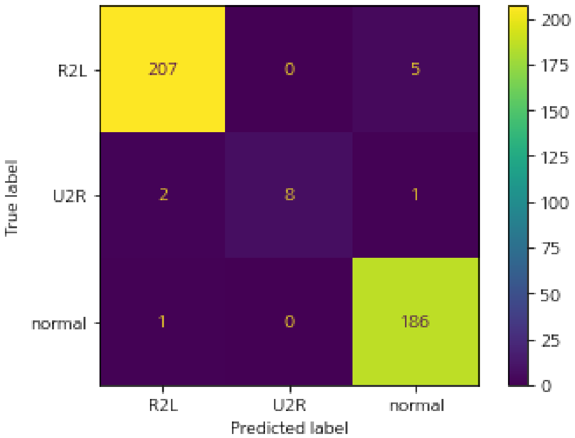

Fig. 3. Confusion matrix of classification results from Random Forest (R2L, U2R) model

In the case of 'RF', the classification accuracy of 'R2L' is 0.9764 and 'U2R' is 0.7272. These results show a higher classification performance than the accuracy of 0.5384 when classifying five types of attacks.

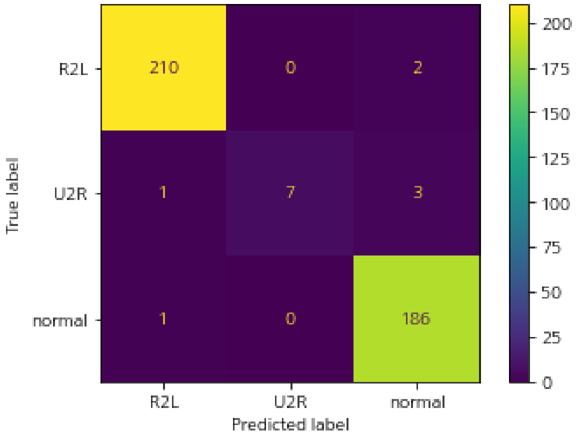

Fig. 4. Confusion matrix of classification results from LGBM (R2L, U2R) model

For 'LGBM', the classification accuracy of 'R2L' is 0.9705 and 'U2R' is 0.6363. In this case as well, the classification performance is higher than the accuracy of 0.5384 when classifying 5 types of attacks. When synthesizing the results, the low classification performance for 'R2L' and 'U2R' observed in the five categories ('Normal', 'DoS', 'Probe', 'R2L', 'U2R') is believed to be due to the failure in adequately classifying 'R2L' and 'U2R', which had not been learned properly due to the issue of class imbalance in the overall dataset

In particular, in the case of 'U2R', the accuracy of the entire data is 0.5384, which is not a significant classification. However, when training with a dataset consisting of 'normal', 'R2L', and 'U2R' as shown in Table 2, we can see that the accuracy of the random forest classifier is 0.7272 and the accuracy of LGBM is 0.6363. The implication of these results is that building a classification model for five attack types may be insufficient to defend against 'R2L' or 'U2R' attack types.

Variable Importance by Model

In order to determine which variables were important when classifying RF and LGBM, the variable importance of the two models is shown in Fig. 5 and 6.

In the case of 'RF', 'service', 'dst_host_srv_count', and 'src_bytes' were identified as important variables in that order. On the other hand, in the case of 'LGBM', 'src_bytes', 'dst_host_srv_count', and 'dst_bytes' were selected as important variables. In the two models showing similar performance, the importance of variables appears quite different. Of course, the order of variable importance does not necessarily determine the order of the decision tree. The selection and branching criteria for variable importance nodes are in the direction of large information gain, and feature importance is simply sorted in the order of the model's classification error. To investigate the determinants of the model's classification decisions, we applied SHAP, a well-regarded model in XAI, to identify the most influential variables.

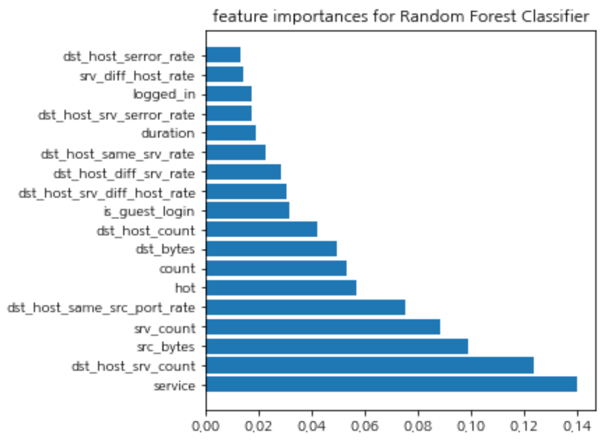

Fig. 5. Feature importances by random forest

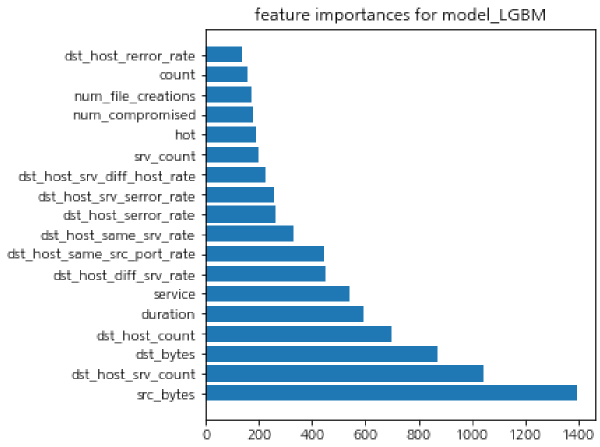

Fig. 6. Feature importances by LGBM

Interpretation of results

Surrogate analysis is an analysis method that determines whether a prototype works by creating a simple substitute that mimics the original function. In XAI, when a model is too complex to be analyzed, it refers to a method of analyzing the original model by creating multiple models that perform similar functions. The biggest advantage of surrogate analysis is that it creates a model that can be explained with a small amount of data regardless of the model, and that surrogate analysis is possible as long as the variables are the same even if the model changes. We used this surrogate analysis technique to check why the classification results of the two models 'RF' and 'LGBM' were derived.

In the case of 'RF', the 'service' variable is used as the most important classification indicator, as shown in Fig. 7. In particular, in the 'U2R' classification with a small amount of data, it is used as the most important variable along with 'dst_bytes' and 'src_bytes'.

On the other hand, in the case of 'LGBM', the 'dst_host_srv_count' variable is used as the most important classification indicator, as shown in Fig. 8. And in the 'U2R' classification, 'dst_bytes' is used as the most important variable. This means that the difference in 'dst_bytes' can be used as an important indicator to determine whether there is a 'U2R' attack. In addition, 'src_bytes' and 'count' are confirmed as important variables in the 'LGBM' model classification results.

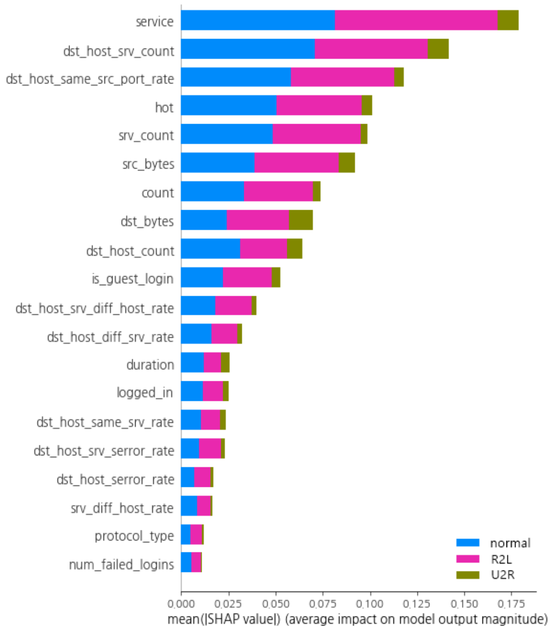

Fig. 7. summary_plot of a specific class by Random Forest

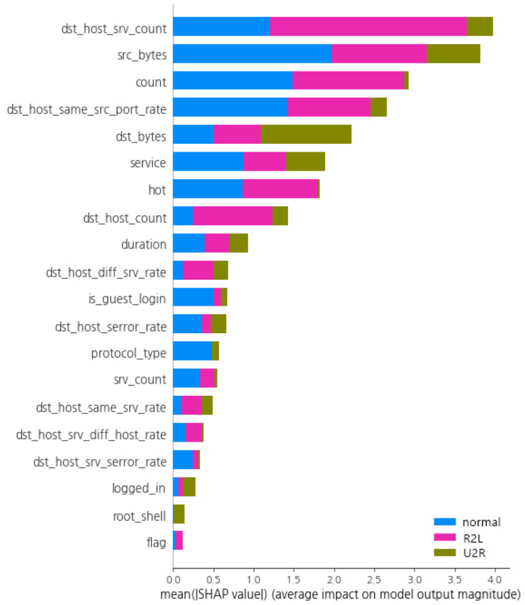

Fig. 8. summary_plot of a specific class by LGBM

Conclusion

In this paper, we confirmed that it is difficult to detect 'R2L' and 'U2R' attacks due to class imbalance in the machine learning model that detects 'DoS', 'Probe', 'R2L', and 'U2R' network attacks. So, as a solution to this problem, we propose a method of learning a separate detection model. As shown in Fig. 9, this is a method of creating a two-layer machine learning model. The first layer detects 'DoS' and 'Probe' type attacks, and the second layer uses models for 'R2L' and 'U2R' attacks. Although our new method takes more time and computer power to learn than the basic single-layer model, it greatly improves how well we can detect 'R2L' attacks. The detection rate has gone up to 0.7272 for RF and 0.6363 for LGBM, from the previous 0.5384.

The two-layer model method proposed in this study is capable of detecting attacks against 'R2L' and 'U2R', which have been shown to have good accuracy for certain attacks in existing studies, but were difficult to detect due to the imbalance of data. This can overcome the limitations of the existing model and have a clearer understanding of the attack method, which can overcome the weaknesses of the model, which can lead to preparation for more diverse disaster situations.

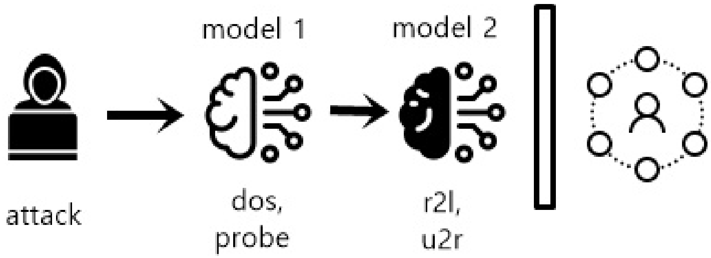

Fig. 9. Intrusion detection with a two-layer machine learning model

We found that 'dst_host_srv_count', 'service', 'src_bytes', and 'count' are important factors for classifying 'R2L' and 'U2R' attacks when we looked into how 'RF' and 'LGBM' models, which are good at detecting these attacks, make their decisions using SHAP analysis. Looking at important variables this way lets us understand what the model is using to classify and shows us which variables to focus on to improve how it works. However, since the number of rows in the 'U2R' dataset is 52, which is 10% of the other datasets, this study cannot be considered to have completely solved the problem of class imbalance, and it is difficult to judge that LGBM is a model that succeeds in learning completely with a recall score of 0.64. To solve these problems More data of 'U2R' is needed, but since it is difficult to obtain data on the actual content of the attack, in further research, we would like to conduct a study to generate 'U2R' using the over sampling technique, which is a method of generating 'U2R' with similar variable values to the data, and compare the performance of the two-layer detection model proposed in this study and clarify the basis of the model's classification for 'U2R' attacks.

References

[1] Bace, R.G., Mell, P. (2001). Intrusion Detection Systems. National Institute of Standards and Technology, NIST Special Publication on Intrusion Detection Systems, Gaithersburg, US.

[2] Jeong, M.K., Lee, S.H., Kim, C.S. (2020). “A study on the safety index service model by disaster sector using big data analysis.” Journal of the Korea Society of Disaster Information, Vol. 16, No. 4, pp. 682-690.

[3] KDD CUP 1999 Data, The UCI KDD Archive, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[4] Khraisat, A., Gondal, I., Vamplew, P., Kamruzzaman, J. (2019). “Survey of intrusion detection systems: Techniques, datasets and challenges.” Cybersecurity, Vol. 2, No. 1, pp. 1-22.

[5] Le, T.-T.-H., Kim, H., Kang, H., Kim, H. (2022). “Classification and explanation for intrusion detection system based on ensemble trees and SHAP method.” Sensors, Vol. 22, No. 3, 1154.

[6] NSL-KDD dataset, University of New Brunswick, http://www.unb.ca/cic/research/datasets/nsl.html

[7] Ribeiro, M.T., Singh, S., Guestrin, C. (2016). ““Why should I trust you?”: Explaining the predictions of any classifier.” Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135-1144.

[8] Shapley, L.S. (1953). A value for n-person games, Contributions to Theory Games, vol. 2, Princeton University Press, Princeton, US.

[9] So, B.G., Jeong, J.S. (2021). “Cyber risk management of SMEs to prevent personal information leakage accidents.” Journal of the Korea Society of Disaster Information, Vol. 17, No. 2, pp. 375-390.

[10] Wali, S., Khan, I. (2021). “Explainable AI and random forest based reliable intrusion detection system.” [online] Available: https://www.techrxiv.org/articles/preprint/Explainable_AI_and_Random_Forest_Based_Reliable_Intrusion_Detection_system/17169080.

[11] Wang, M., Zheng, K., Yang, Y., Wang, X. (2020). “An explainable machine learning framework for intrusion detection systems.” IEEE Access, Vol. 8, pp. 73127-73141.

[12] Wang, Y., Wang, P., Wang, Z., Cao, M. (2021). “An explainable intrusion detection system.” IEEE International Conference on High Performance Computing and Communications (HPCC), 2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), Haikou, Hainan, China, pp. 1657-1662.