A Machine Learning Driven Intrusion Detection Model Using Logistic Regression and Transfer Learning

Promise Enyindah[1]; Umejuru Daniel[2]

[1,2]Department of Computer Science, University of Port Harcourt, Rivers State, Nigeria.

[1]https://orcid.org/0000-0001-6246-7077
[2]https://orcid.org /0009-0007-9843-2248

Publication Date: 2026/01/24

Abstract: Network intrusion detection systems (IDS) play a critical role in protecting modern network communications by analyzing patterns in network traffic to identify potential attacks and policy violations. Recent advances have seen IDSs leverage both traditional and deep machine learning techniques, though developing such models often demands large datasets, extensive computational resources, and multiple training iterations. This study presents a network intrusion detection approach based on transfer learning, aiming to improve detection efficiency while reducing the cost and complexity of model training. Two pre-trained convolutional neural network (CNN) models were adapted for IDS tasks using knowledge transfer, enabling the integration of predictions into a single enhanced model. The system was trained and evaluated using the NLS-KDD benchmark dataset, covering normal traffic as well as probing, Denial-of-Service (DoS), user-to-root (U2R), and remote-to-local (R2L) attack types. Experimental results show that the transfer learning approach achieved a prediction accuracy of 96.52%, significantly outperforming a traditional logistic regression model, which achieved 66.56%. These findings demonstrate that transfer learning can effectively enhance IDS performance, improving both reliability and accuracy in detecting diverse network threats.

Keywords: *Logistic Regression, Intrusion, Detection, Attack, Transfer Learning.*

How to Cite: Promise Enyindah; Umejuru Daniel (2026) A Machine Learning Driven Intrusion Detection Model Using Logistic Regression and Transfer Learning. *International Journal of Innovative Science and Research Technology*, 11(1), 1814-1825. https://doi.org/10.38124/ijisrt/26jan777

I. INTRODUCTION

Recent technologies that used artificial intelligence to resolve problems in huge data and the internet of things have transformed our lives progressively more reliant on the internet (Vadhil, et al. 2024). In addition, the frequency of unusual behaviors is growing. Identifying abnormal network behavior is a key IDS challenge that is growing increasingly important, particularly because we rely more on laptops and cell phones in recent years (Singh et al.,2022). Our everyday activities are transferring to the World Wide Web, making safety challenges more complex than ever because of the worldwide pandemic (Albulayhi et al.,2015). To identify unusual actions in a personal computer or network, there is a particular surveillance device known as network intrusion detection system. There are different kinds of intrusion detection systems (IDS) that are designed to detect, alert, or identify intruders attempting to gain illegal access to a network. In digital technological advances, intrusion detection systems (IDSs) can learn or see unwanted access to a networking infrastructure or environment (Singh, et al.,2022). This intrusion detection system in a shell not only secures the networking environment but also serves as a preventive measure (Diogenes et al. 2023). Networking possesses a lot of knowledge embedded in it and is intended to be secured through all means; however, nothing is completely safeguarded and novel instances are always discovered as developments in technology occur; that is why the necessity of an intrusion detection system arises, generally, in the information technology environment (Solanki, et al.,2020). The attribute that needs collection procedures and feature patterns differ across data items; hence the best ML models also differ. According to Parag et. al (2021) the hierarchy of intrusion detection systems comprises the following features: (a.) features that can best represent different attack severity levels. (b.) the kind of data best suited for detecting specific attacks (c.) ML methods that are best suited to certain data type (d.) How do machine learning technologies strengthen IDS through various ways? Intrusion detection is a global infrastructure for the information society, enabling advanced services based on existing and evolving interoperable information and communication technologies (Sheikh et al. 2022).

Intrusion detection system is a system designed to detect, to notice, or to spot an intruder, trying to gain an unauthorized access into a system. In information technology, it is a system to notice, or spot an unauthorized access into a networking system or environment. This intrusion detection system in a networking environment does not include, in general, a preventive measure. Networking today, has a lot of information embedded in it, and is supposed to be protected by all means, but nothing is 100 percent protected; that is why the need for an intrusion detection system arises, generally, in the information technology environment.

Wireless Sensor Networks (WSNs) continue to grow as one of the most exciting and challenging research areas of engineering. There are many applications of WSNs which are intended to monitor physical and environmental phenomena such as ocean and wildlife, earthquakes, pollution, wild fires and water quality. WSNs can also be used to gather information regarding human activities such as health care, manufacturing machinery performance, building safety, military surveillance and reconnaissance, highway traffic, and more (Butun, 2023).

As science and technology grow by the day, security is becoming a major issue and everybody's concern. Security network protocol designers are trying to make security stronger as a part of its (networking), and as well easy to use. There are certain network security measures and protocols that are supposed to be put in place for a better security check and protection against unauthorized access. Such protocols are evolving as science and technology grow with new network scanning, network mapping, network searching, and network analyzing algorithms, giving birth to new and useful networking tools, modules, and software, including hardware.

There has been series of inventions and development on the area of security networking as it relates to information sharing and gathering. Various software applications are used for security issues corrections like, malicious software (Malware) application, Antivirus application, Trojan horse removal application, as well as other software for more technical applications like scanners. Such scanners are Network Mapper (NMap) tool, WireShark tool, Universal Serial Bus (USB) scanners and monitors. The invention of the so many operating system is a welcome networking invention to combat networking issues, specifically on data and information sharing and gathering, and also with the use of manually activated over 600 built-in security tools.

But still, there is a unending challenges in the networking environment due to some network users launching attacks on cooperate and individual networks with the use of malicious software and codes with the aim of stealing information (or data), destroy the networking system, hijack the networking system, or even upload their information (or data) to the network which is aimed at blackmailing and defrauding the network owners (Chawla 2023).

II. CONCEPTUAL FRAMEWORK

In modern era, organizations greatly admit computer networks to share info throughout the organization in associate degree economical and productive manner. Structure computer networks square measure currently changing into giant and omnipresent. Presumptuous that every employee contains a dedicated digital computer, an outsized scale company would have few thousand workstations and plenty of server on the network.

It is doubtless that these workstations might not be centrally managed, nor would they need perimeter protection. They will have a spread of operative systems, hardware, software, and protocols, with completely different level of cyber awareness among users. Currently imagine these thousands of workstations on company network square measure directly connected to the net. This type of unsecured network becomes a target for associate degree attack that holds valuable info and displays vulnerabilities.

Sources of potential security problems are challenges and attacks, while the risk relates to the probable outcome and its associated costs due to occurrence of certain events. There are numerous techniques help protect your computer: cryptography, authentication, checked the software, licenses and certificates, valid authorization (Ambusaidi, 2022).

Network security is a broad term that covers a mass of technologies, devices and processes. In its simplest term, it's a group of rules and configurations designed to shield the integrity, confidentiality and accessibility of computer networks and information mistreatment each software package and hardware technologies. Each organization, in spite of size, trade or infrastructure, needs a degree of network security solutions in situ to shield it from the ever-growing landscape of cyber threats within the wild these days.

Today's spec is complicated and is long-faced with a threat atmosphere that's continually ever-changing and attackers that are continually attempting to search out and exploit vulnerabilities. These vulnerabilities will exist in a very broad variety of areas, as well as devices, data, applications, users and locations. For this reason, there are several network security management tools and applications in use these days that address individual threats and exploits and additionally regulative non-compliance. Once simply some minutes of period will cause widespread disruption and big injury to associate degree organization's bottom line and name, it's essential that these protection measures are in place.

It is evident that societal issues like cybersecurity need to be addressed by different parties, such as Internet service providers, telecom organizations and governmental agencies. However, it is equally important that end users behave in a secure fashion, as they play an essential role in safeguarding the online domain. Moreover, they are essential for achieving online security (Furnell et al, 2024).

➢ *Classification*

This is the process of assigning data items to pre-defined classes. The result of this process will be a classifier based on association rules or decision trees. For example, suppose sufficient "normal" and "abnormal" audit data is gathered for a user or a program, then a classification algorithm is applied to learn a classifier that can label or predict new audit data as belonging to the normal class or the abnormal class.

Classification categorizes the datasets into pre-defined classes. There are two steps, i.e., training and prediction. In the first step, classifier is trained by analyzing a training set made up of data instances and their associated class labels. Because the class label of each training instance is provided, this is known as supervised learning. In the second step, the trained classifier is used to predict the class for unlabeled data instance (Ghosal and Halder, 2022). An algorithm that implements classification is known as a classifier. The term "classifier" sometimes also refers to the mathematical function, implemented by a classification algorithm that maps input data to a category. The classes are pre-defined in training phase. In terms of the predefined classes, classification may have two cases – Binary and Multiclass classification. in binary classification, only two classes are involved, whereas multiclass classification involves assigning an instance of dataset to one of several classes (Hu Y, et al., 2020).

In principle, classification may be used for both misuse detection and anomaly detection, but mostly used for misuse detection. As for misuse detection, the detection can be formulated as a classification problem. Suppose sufficient audit data has been gathered in which each data instance will be labeled as either “normal” or “abnormal”. We then use classification algorithm on audit data to train a classifier.

➢ *Transfer Learning (TL)*

Transfer learning (TL) is a new approach that allows models to use knowledge from similar tasks, categories, or pre-trained models. TL is a method used in machine learning that involves adapting a model trained on a single endeavor to a comparable task (Mohammad, et al., 2021). In conventional machine learning, representations are trained separately for every task. Transfer learning, on the other hand, enables models to apply knowledge from a previous task in order to increase performance on a subsequent one.

➢ *Logistic Regression (LR)*

LR, also known as the logistic or logit model, is a tool for examining the relationships underlying categorical variables that are dependent and a large number of independent factors. It also determines the likelihood that an event will occur by matching data to a logistic curve.

III. METHODOLOGY

This study will be done utilizing the Object-Oriented Analysis and Design Methodology (OOADM). The purpose is to understand, model, and build the proposed system as a collection of interconnected classes and objects. The OOADM is a technological method for analyzing and designing an application, system, or business that employs object-oriented programming and visual modeling throughout the software development process to affect stakeholder communication and product quality. Additionally, OOADM has exhibited flexibility and adaptability in the ever-changing software market. The unified model, combined, and agile approaches are the next phases in the evolution of the object-oriented technique from relatively structured and object-oriented design (Khan et al., 2011).

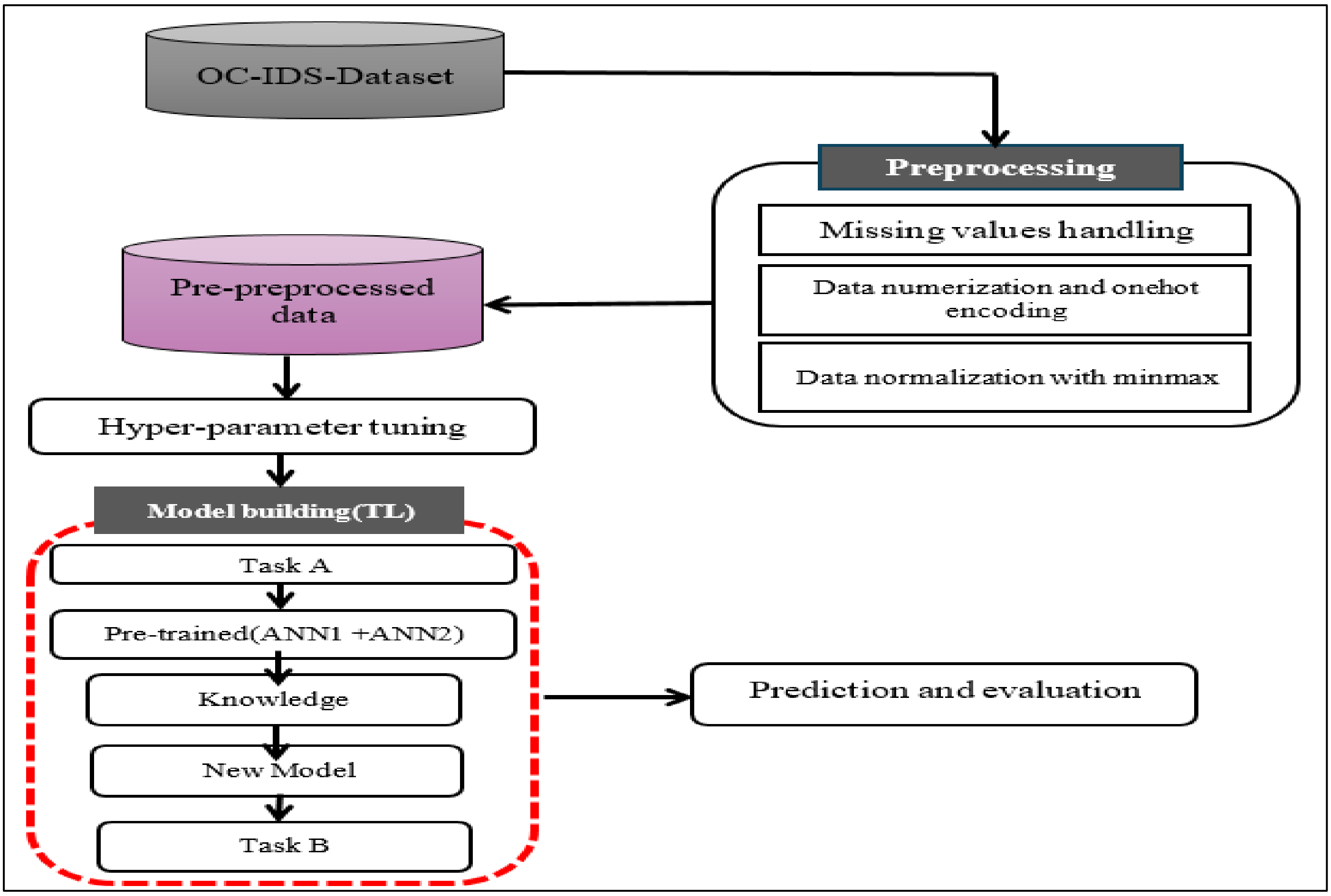


Fig 1 Proposed System Architecture

➢ *Analytical Presentation of Components of the Proposed System*

- *Datasets:*

The dataset was taken from the Kaggle site and contains the attributes of an IoT IDS dataset recorded in CSV excel format. The suggested collection offers a big dataset for developing the proposed model, with about 6763 testing and 15,780 training sets comprising the 22,543 datasets. For this experiment, we are using the IOT IDS dataset retrieved from the Kaggle website, which has 22,545 items. It contains 12833 IDS assaults and 9711 normal cases with 42 features (protocol, services, number of logins, number of failed attempts, and so on), as shown in the table.

- *Preprocessing:*

Preprocessin is a step containing feature engineering and scaling to transform data in a way that algorithm can lean and understand pattern in data. The collection contains IDS attack dataset stored in excel csv file format. We applied data preparation technique like feature scaling to better understand the IDS data and features).

➢ *Classification Technique*

The equations are an exception to the prescribed is one of the key and useful components involved in decision making process that categorize data based on some observed features or criteria. We employed the feature extraction technique in classification where a feature vector(X) in the process represented by as:

$$X = (f_1, f_2, \ldots, f_n) \qquad (1)$$

Where "f" represents the features and "n" number of features and efficiently classified to form an appropriate class. We adopted row and class-feature sampling technique for each and every decision tree to reduce bias and high variance. The Change in input dataset causes low variance in the decision tree and output will be very good and accurate with majority votes for the binary classification model.

$$\Delta_k = \{(P_1, \ldots, P_k): \textstyle\sum_{k=1}^{N} P_k = 1 \; and \; P_k \geq 0\} \qquad (2)$$

Where Δk is the set probability distribution over X, we therefore assume that ek to be a member or element of Δk and if a decision-tree(t) predicts that an instance to a class Xk

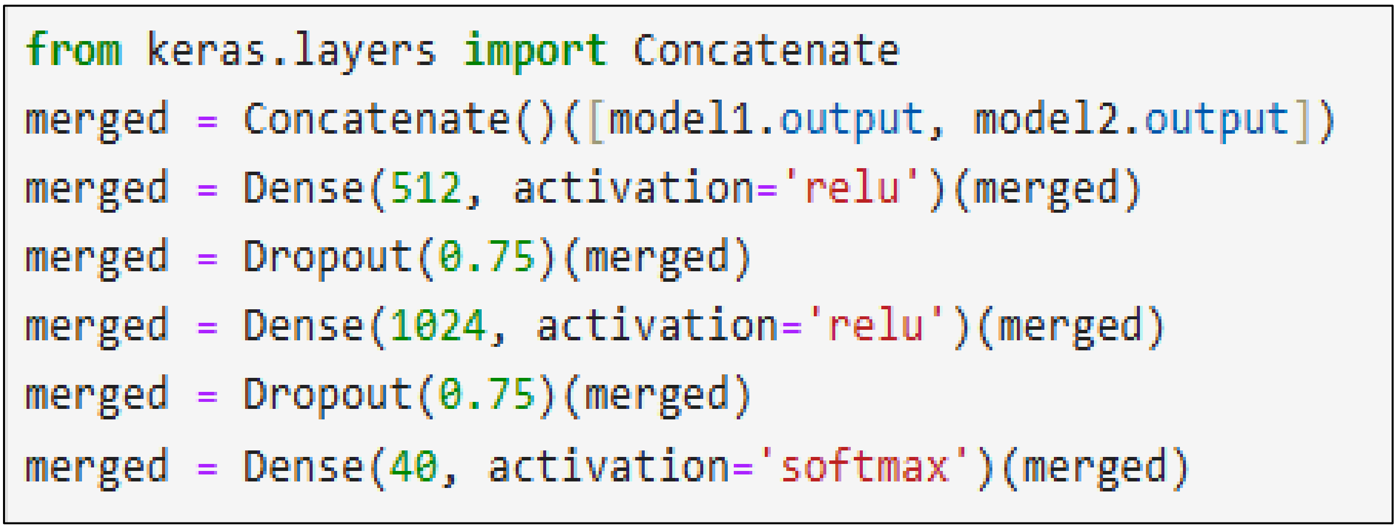
- *Training and Testing Dataset:*

The uploaded dataset of the proposed system is divided into 80% (1437 items) and 20% (360 items) of the total dataset (1797 items) for training and testing using the xtrain, xtest, ytrain, ytest = train _and_test_split and test_size set to 0.2 command. The sklearn and ensemble libraries are employed to build multiple decision trees in predicting the outcome based on majority vote. To train and evaluate the mode we employed the training (80%) and testing (20%) of the total dataset.

➢ *Steps Required in Building Transfer Learning in ANN Models*

A machine learning (ML) technique called transfer learning (TL) involves fine-tuning a model that has already been trained on one task for a new, related task. TL is a machine learning (ML) technique that involves.

- A pre-trained component, that is, a ML model that has been trained on a large dataset and can be fine-tuned for a given task A. It is frequently used as a starting point for constructing ML models since they contain a set of initial weights and biases that may be fine-tuned for a specific task.
- Knowledge in TL refers to the information, patterns, and abstractions that a ML model acquires when training on a certain task. This knowledge can be used to enhance performance in an entirely different but related activity or topic.
- New model: A new model is built by employing the components of a model that was previously trained for a different but similar task: Transfer learning initializes a new model with feature representations from a previously trained model. This enables us to avoid training a new model from begin.
- Obtain predictions from the pre-trained models: The set of parameters that are fed into the pre-trained model in the same order as the original dataset can then be predicted using the pre-trained model as shown in the screen shot given below:



```
from keras.layers import Concatenate
merged = Concatenate()([model1.output, model2.output])
merged = Dense(512, activation='relu')(merged)
merged = Dropout(0.75)(merged)
merged = Dense(1024, activation='relu')(merged)
merged = Dropout(0.75)(merged)
merged = Dense(40, activation='softmax')(merged)
```

Fig 2 Pre-Trained Model

Table 1 Logistic Regression

Algorithm 1: Logistic Regression	
Step	**Processes Involved**
1	Start
2	Define LR regression model
3	Select logistic algorithm to measure IDS attacks
4	Initialize residuals
5	Compute value for adjustable response
6	Search and update learned patterns
7	Compute the length training set for each step
8	Display approximated solution
9	Update residuals
10	Stop

Table 2 Transfer Learning in Neural Networks

Algorithm 2: Transfer Learning in Neural Networks	
Steps	**Procedure**
1	Select a pre-trained model (choose a model to use as the base for training which depend on the task)
2	Create a base model (instantiate the base model using an architecture like CNNs)
3	Freeze neural network layers (present weights in the pre-trained model from being re-initiated)
4	Add new trainable layers
5	Train new model layers

IV. RESULTS AND DISCUSSIONS

In this section, the results of the existing LR and transfer learning models are presented and discussed using heat maps, confusion matrix, Bar charts and tables. To improve the classification result, different fine-tuned hyper-parameter values were used during the design and implementation process. The confusion matrix, ROC, and classification report provided below are used to illustrate and discuss the prediction and classification accuracy of both models

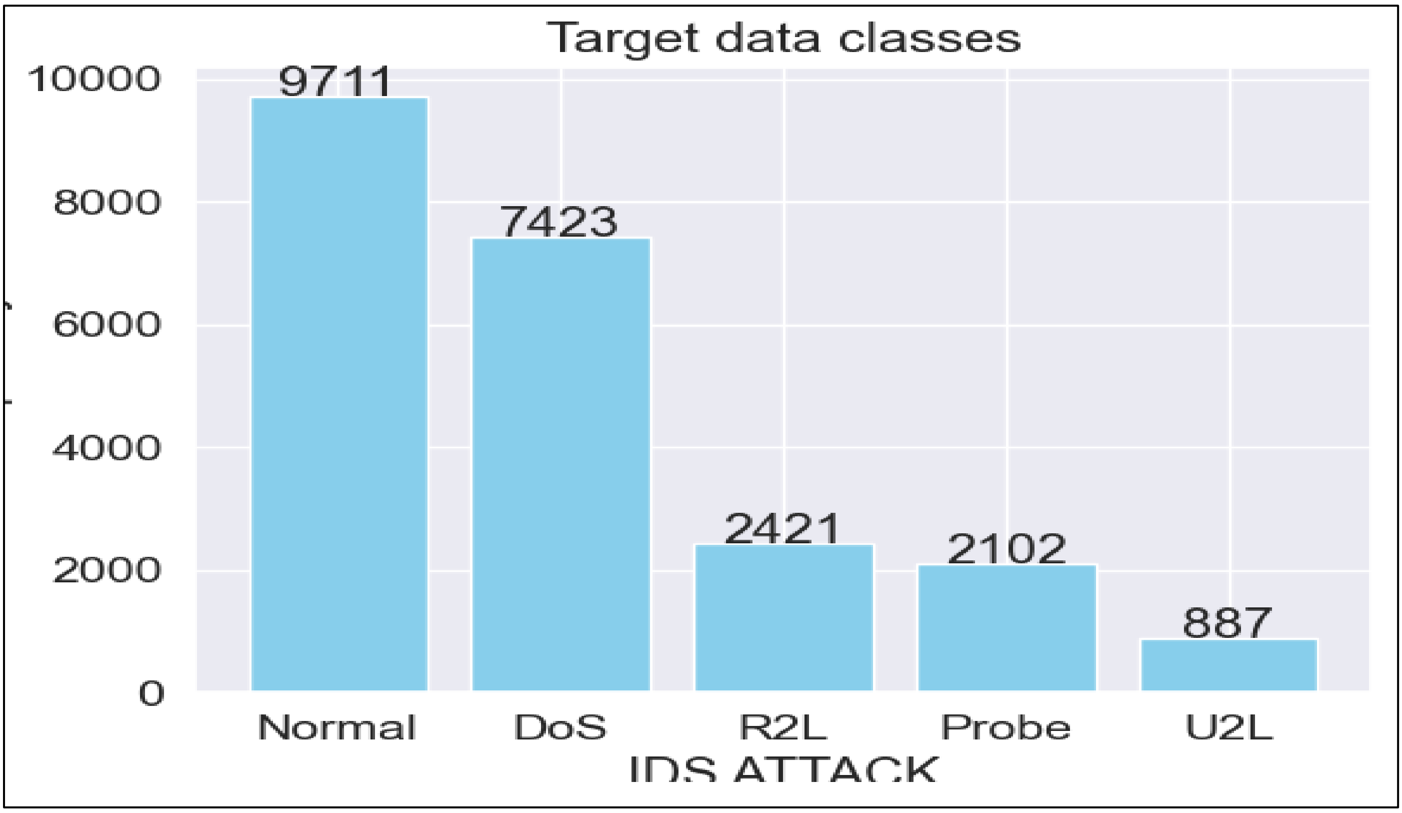


Fig 3 IDS Attack Types in Target Set

Fig 3 displays the number of IDS attack types, including normal instances, Dos, R2L, Probe, and U2R attacks. The normal occurrences recorded 9711 cases, which was significantly more than 800, while the DoS yielded 7423, R2L (2421), Probe (2102), and U2L produced 887 instances. The chart depicts an imbalance in attack target class values across normal, DoS, R2L, U2R, and Probe scenarios.)

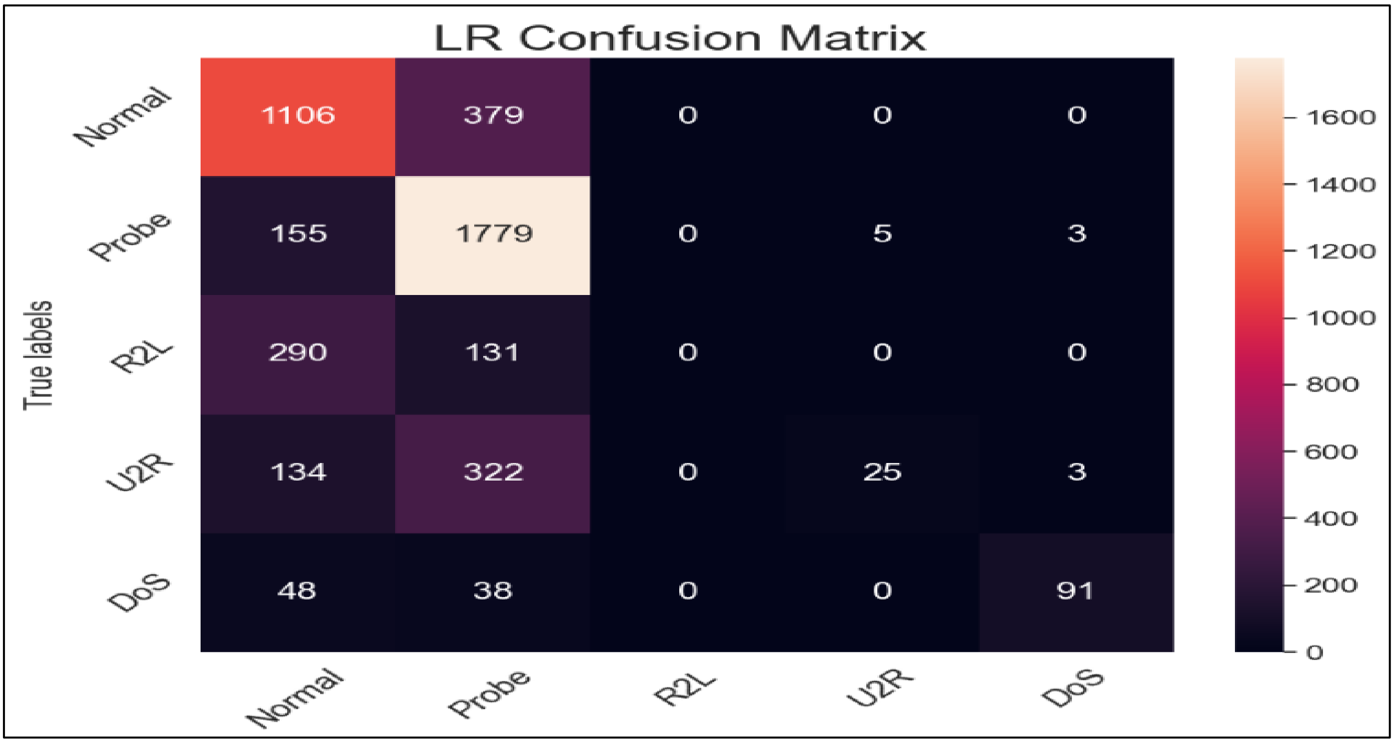


Fig 4 LR Confusion Matrix

Fig 4 depicts the confusion matrix of existing logistic regression. According to the confusion matrix, the overall number of correct predictions is recorded in the main diagonal = 1106 + 1779 + 25 + 91 = 3001 correctly classified cases, while the total number of incorrect predictions is recorded above and below the main diagonal = 279 + 3 + 155 + 290 + 131 + 134 + 322 + 48 + 38 = 1397 incorrect cases. There are more misclassifications in the existing LR model than correct classifications. The LR model favors the majority target class, which is the normal class, but fails to correctly categorize most attack types into the groups they represent)

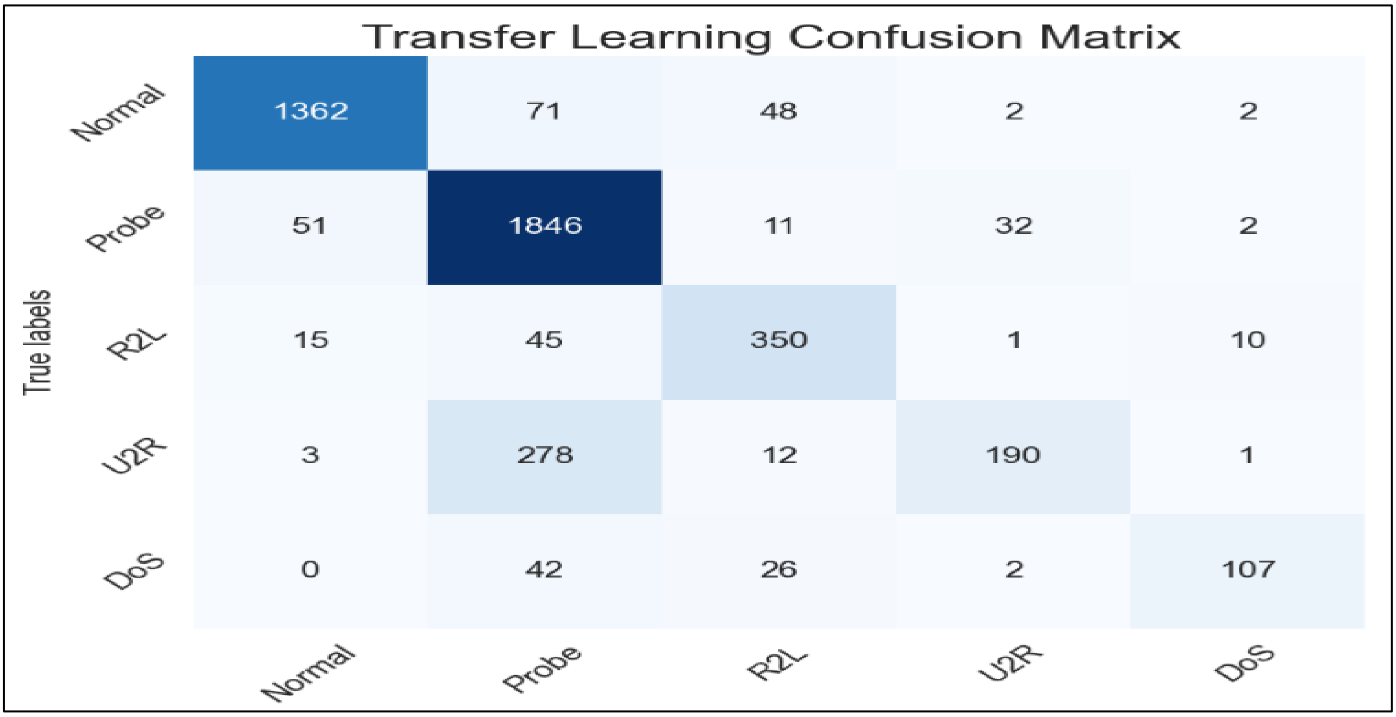
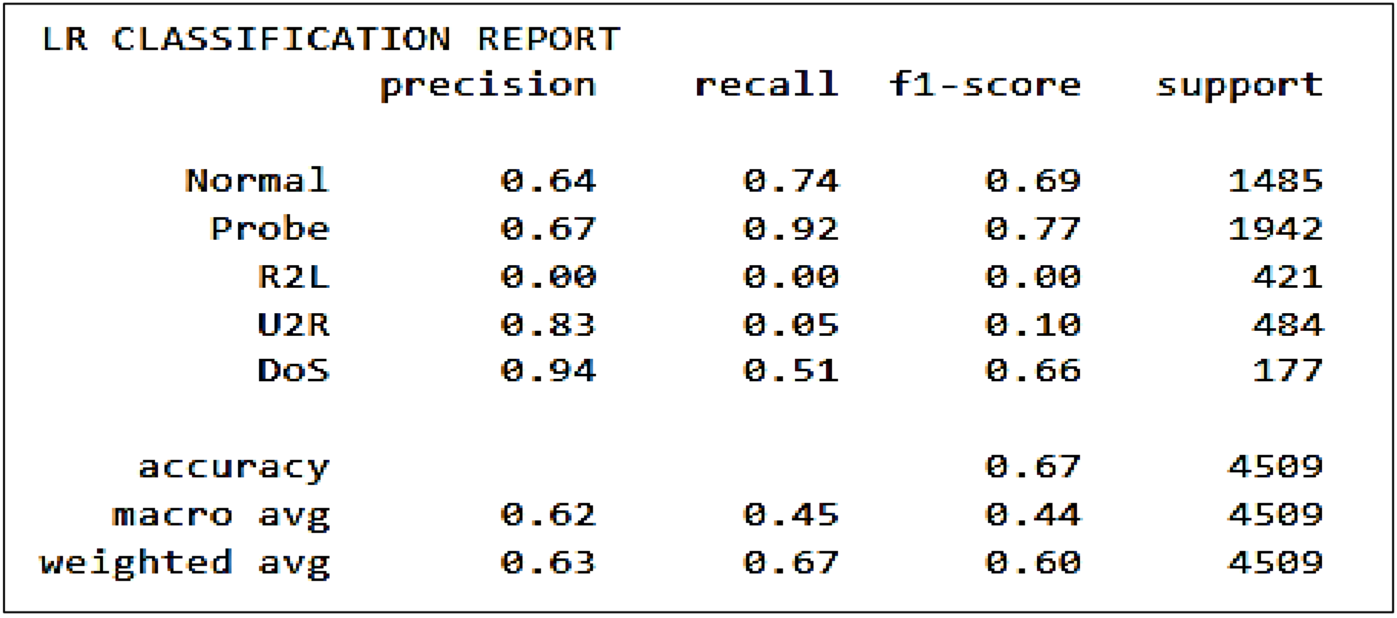


Fig 5 Transfer Learning (TL) Confusion Matrix

Fig 5 shows a 4x4 confusion matrix for the TL model, indicating the number of targeted classes used to evaluate multi-classification performance. This is done to compare the predicted outcomes for the transfer learning model to the actual goal values. According to the data, 1362 + 1846 + 350 + 190 + 107 = cases of correct classifications and (71 + 48 + 2 + 2 + 11 + 32 + 2 + 1 + 10 + 1) + (51 + 15 + 45 + 3 + 278 + 12 + 42 + 26 + 2) = 180 + 474 = 654 classes of wrongly classified instances were recorded above and below the main diagonal. The confusion matrix helped to explain how well a classification system performs on a set of experimental data for predicting the target class for normal, DoS, probing, and other types of IDS attacks.

Table 3 LR Classification Report



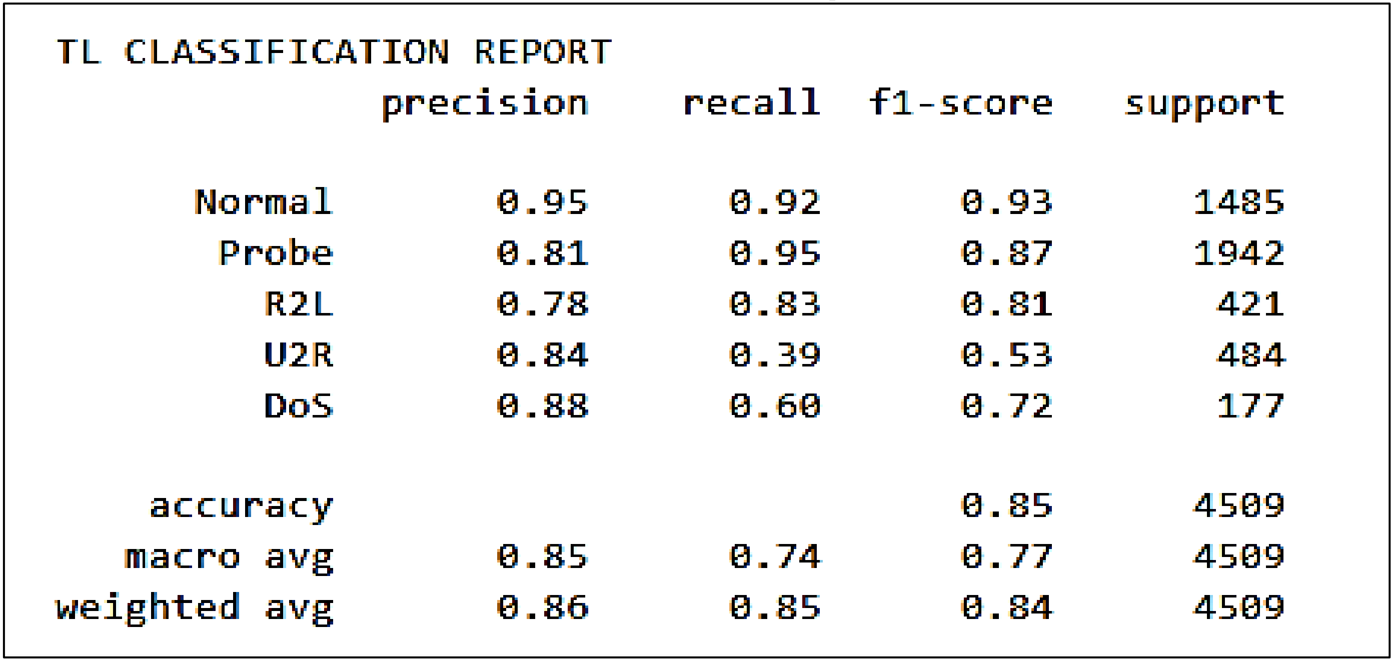
```
LR CLASSIFICATION REPORT
              precision    recall  f1-score   support

      Normal       0.64      0.74      0.69      1485
       Probe       0.67      0.92      0.77      1942
         R2L       0.00      0.00      0.00       421
         U2R       0.83      0.05      0.10       484
         DoS       0.94      0.51      0.66       177

    accuracy                           0.67      4509
   macro avg       0.62      0.45      0.44      4509
weighted avg       0.63      0.67      0.60      4509
```

Table 3 displays the classification report of LR algorithm, with attributes precession, recall, and f1-score classification accuracy of 0.64, 0.74 and 0.69 for normal cases. Probe attack type gave 0.64 (precision), 0.74 (recall), and 0.69 (f1-score). R2L attack yielding 0.00 for precision, recall and f1-score, U2R produced 0.83 (precision), 0.05 (recall), and 0.10 (f1-score) while DoS attack recorded 0.94 for precision, recall (0.51) and f1-score (0.66). The LR provided an accuracy of 0.67 with an average weighting of 0.63 for precision, recall (0.45), and f1-score (0.60) rates.

Table 4 TL Classification Report



```
TL CLASSIFICATION REPORT
              precision    recall  f1-score   support

      Normal       0.95      0.92      0.93      1485
       Probe       0.81      0.95      0.87      1942
         R2L       0.78      0.83      0.81       421
         U2R       0.84      0.39      0.53       484
         DoS       0.88      0.60      0.72       177

    accuracy                           0.85      4509
   macro avg       0.85      0.74      0.77      4509
weighted avg       0.86      0.85      0.84      4509
```

Table 4 is the classification report of TL with precession, recall, and f1-score classification metrics for IDS attacks. For normal cases, accuracy (0,95), recall (0.92), and f1-score (0.93). Probe attacks had a rate of 0.81, R2L (0.78), U2R (0.84), and DoS yielded 0.88 in precision, recall, and f1-score. There is a significant improvement, as shown in the precision, recall, and f1-score values from the classification report. The macro-average shows how all categories equally contributed to the final averaged metrics, the weighted-average shows how each class appears to contribute to the average as weighted by its size, and the micro-average clearly demonstrates how all samples equitably make a contribution to the final averaged metrics.

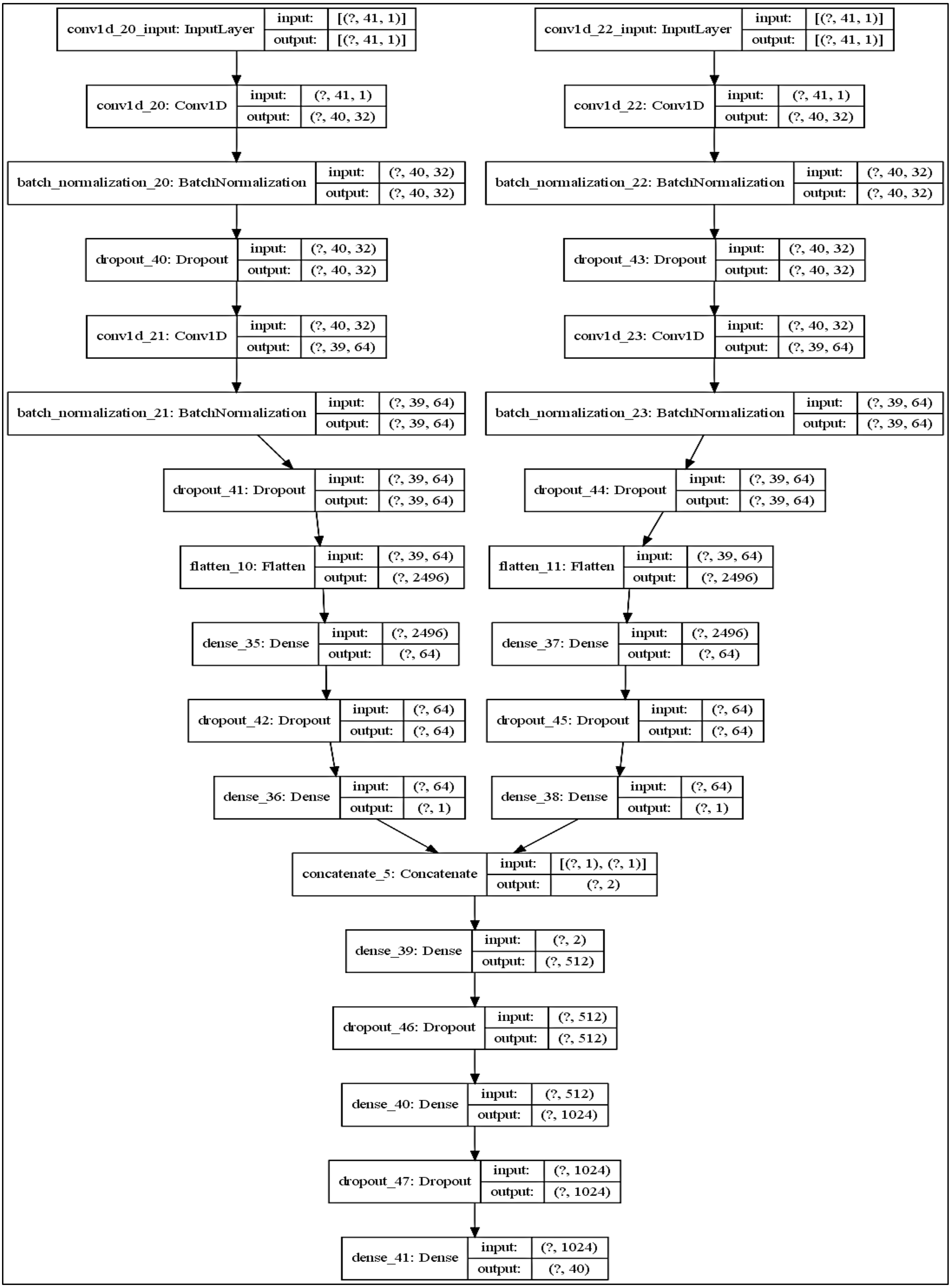


Fig 6 Transfer Learning (TL) Architecture

Fig 6 displays the transfer learning implementation of integrating two distinct Neural Networks, which utilizes the transfer learning or knowledge of two or more techniques to generate a model that outperforms the separate learning models. The knowledge acquired by the two CNN (pre-trained) models is transferred to the base or meta-learning model. At the base level, the two CNN learners are combined to form an intermediate prediction model, with one prediction for each learning model that learns from intermediate patterns involving the same target variable. It increases overall performance and typically outperforms individual intermediate models at the base level, as demonstrated in Figure 4.6. The data set for training is divided into k-folds cross validation and fitted using the basic models on the k--1 path of the entire training set to calculate its performance using the test dataset with predictions for the k-th portion. This process is repeated, and the predictions from the training set are used as features to train the ensemble model and visualize the testing dataset. In order to generate the final predictions, three different neural network models are stacked on top of the base model, known as the meta-learning model. The connections between the neural network ensembles use concatenated operations. The meta model at level-1 is a meta-layer that accepts output from the base models (0-level) as new training data.

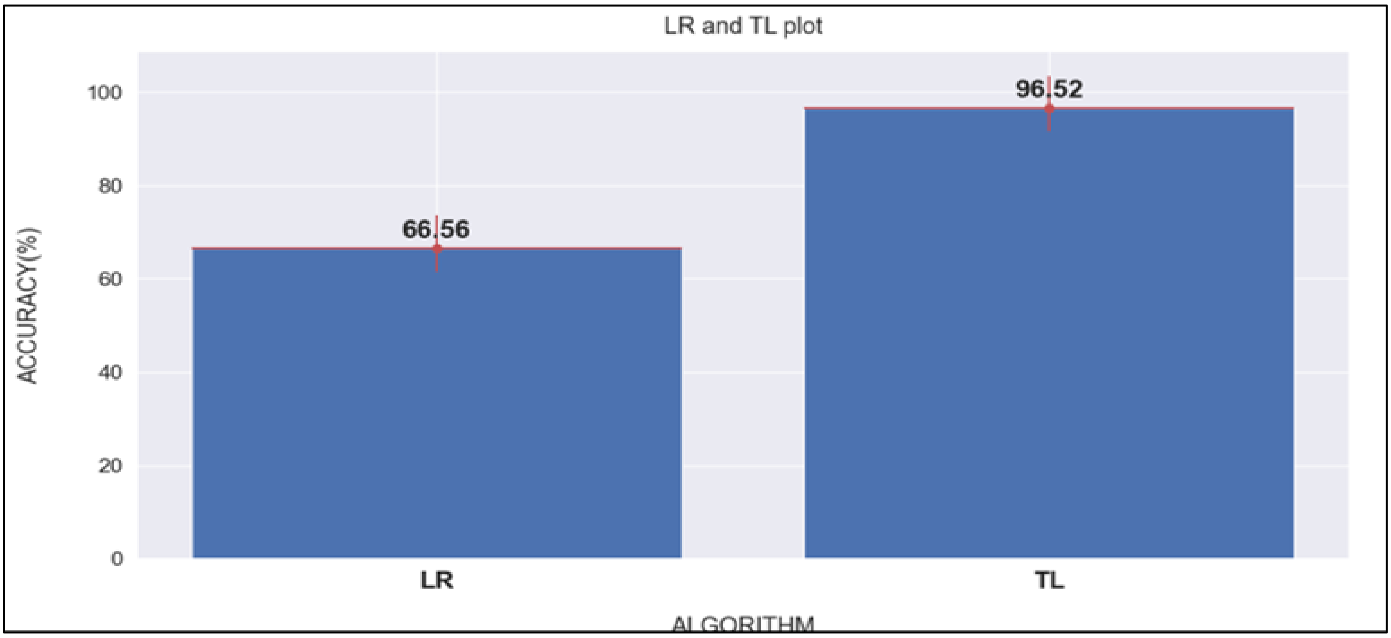


Fig 7 Detection Accuracy of LR and TL

Figure 7 shows the detection accuracy of LR and TL techniques. The TL performed better yielding 96.52% detection accuracy as compared to LR that gave 66.56%. The knowledge transfer idea eliminated model over-fitting and improved the performance of the proposed model by normalizing some activations and increasing the number of network neurons, resulting in increased prediction accuracy.

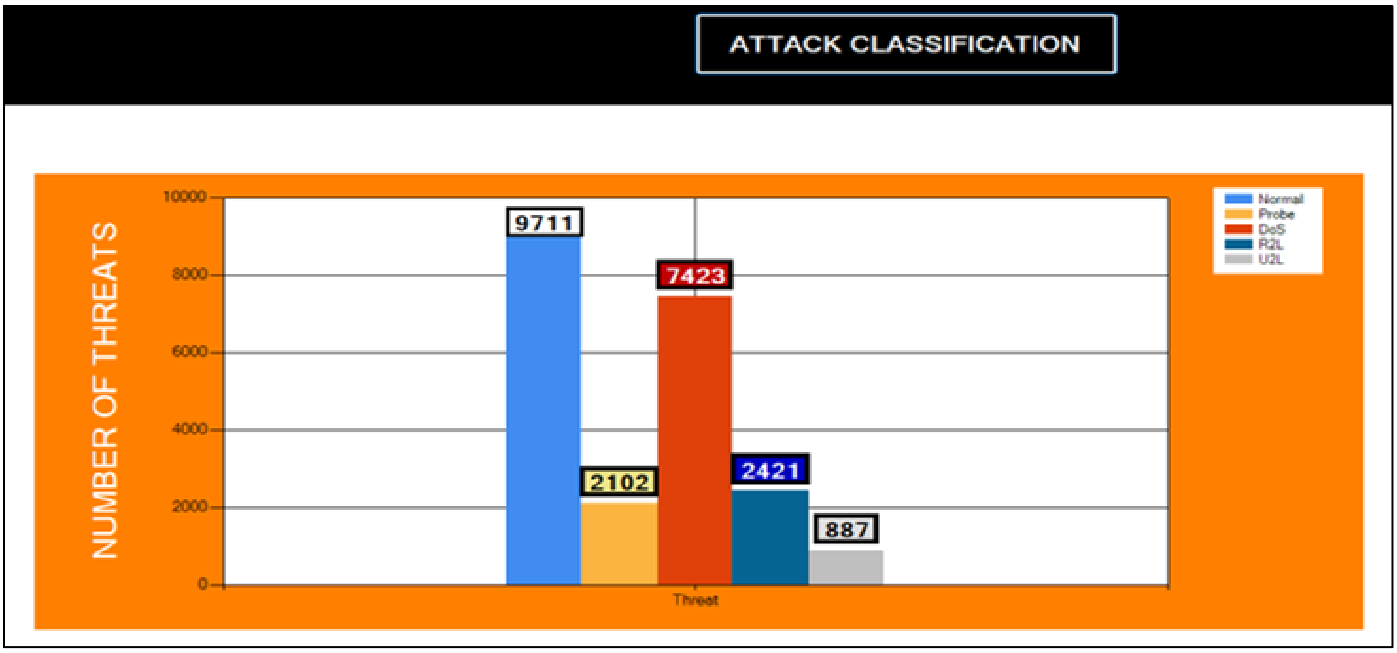


Fig 8 Attack Detection Form

Fig 8 shows the chart generated when a user selects the detect icon before uploading the excel dataset. The "detection" icon depicts normal instances, DoS, probe, U2R, and R2L attack types discovered in the proposed system dataset. The normal instances yielded 9711, probe attack yielded 2102, DoS produced 7423, R2L (2421), and U2R generated 887 cases, as collected from the dataset.

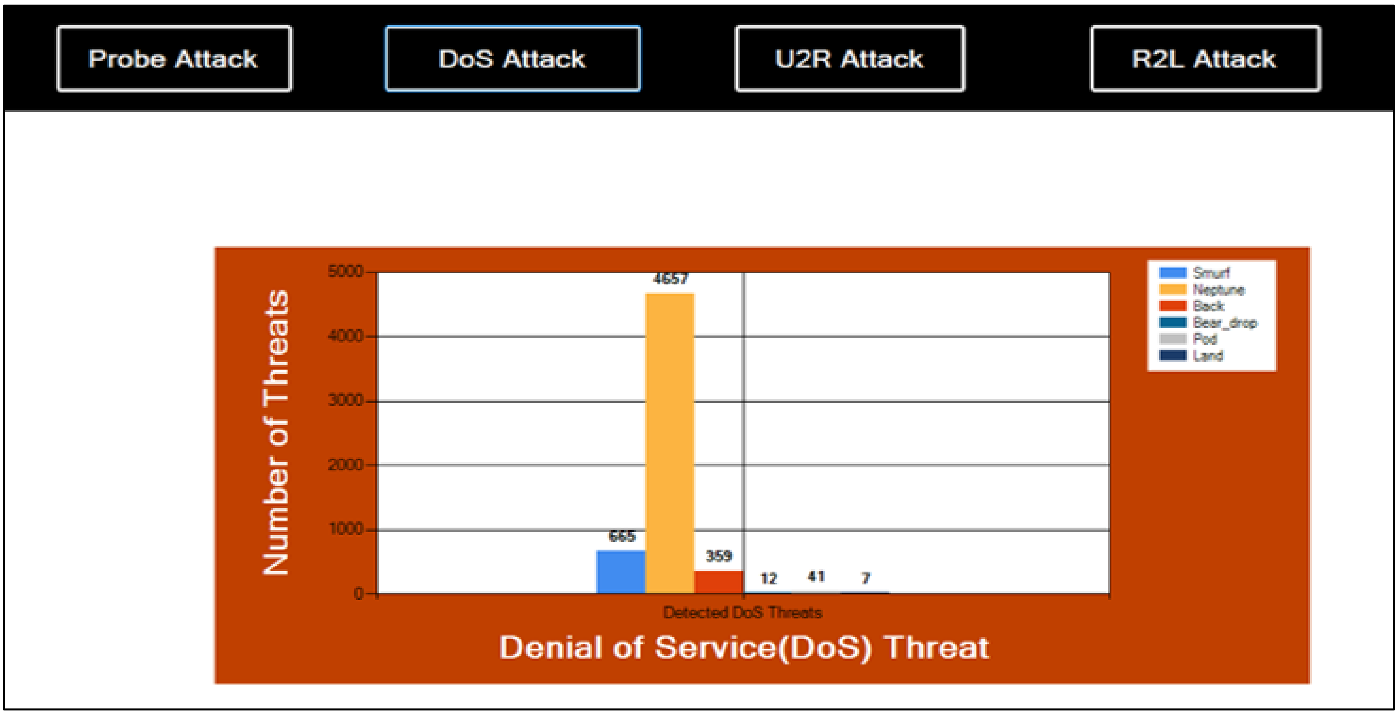


Fig 9 DoS Attack Type Detection Form

Fig 9 depicts the many types of denial-of-service (DoS) attacks, which serve as an unlawful attempt to interrupt or shut down the normal operation of a targeted server, or to render the system or server unavailable to the user. The different types of DoS attacks include Smurf, Neptune, back, bear_drop, pod, and land attacks. The system discovered 665 cases of smurf attacks, with Neptune (4657), back (359), bear drop (12), pod (41) and land DoS attack types accounting for 7 incidents. Neptune DoS attacks had the greatest number of cases, whereas bear drop threats had the fewest in the suggested system dataset.

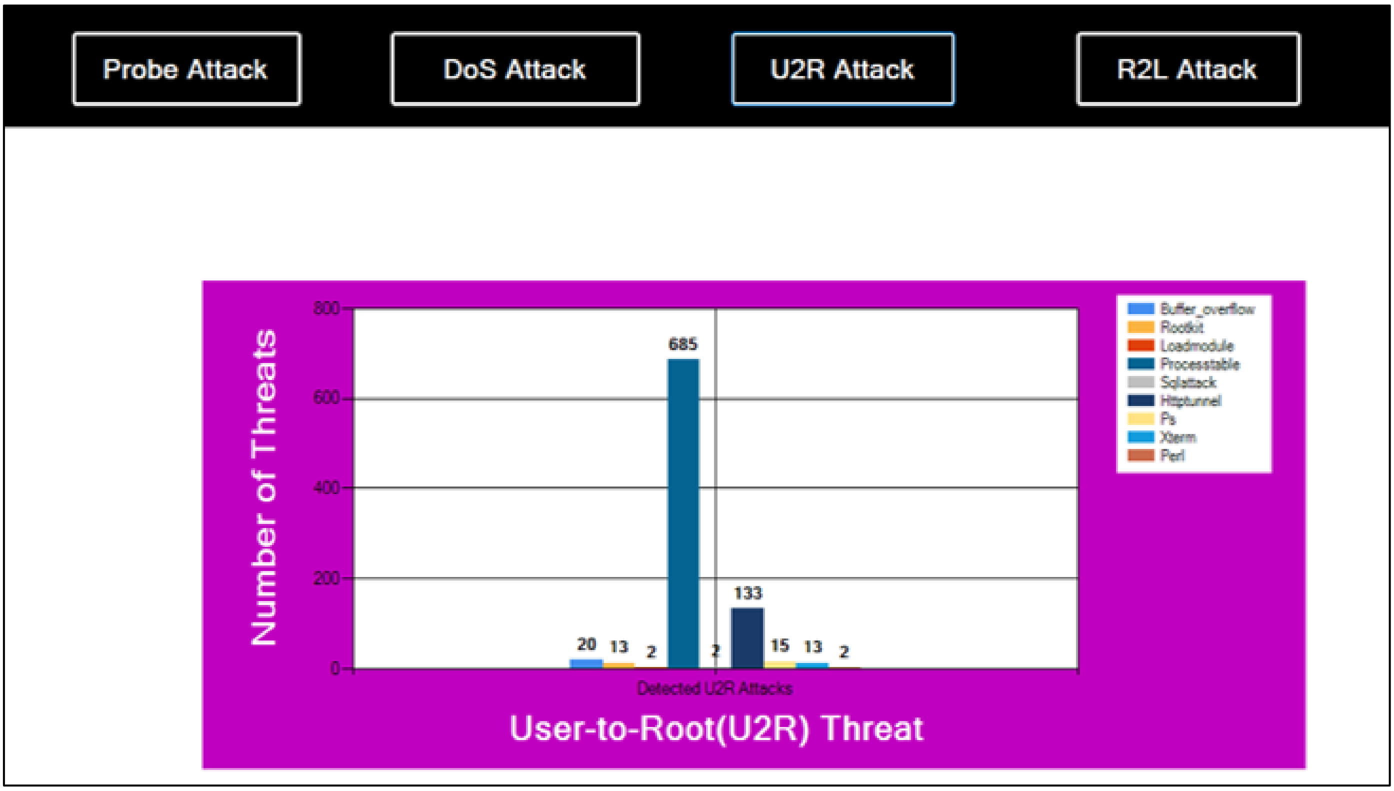


Fig 10 U2R Attack Type Classification Form

Fig 10 depicts the various sub-categories of threats organized into uer-to-root (U2R) sorts of assaults, which include buffer overflow, rootkit, loadmodule, sqlattack, httptunnel, ps, xterm, and perl. The buffer overflow is a U2R threat exploited by cybercriminals to obtain unauthorized access to company networks, with 20 reported occurrences. Hackers adopt a variety of strategies to gain control of a system, including rootkits, which influence the choice of attack vector, yielding 13 possibilities. SQL attack had the most attack incidences, followed by ps, while perl recorded the least.

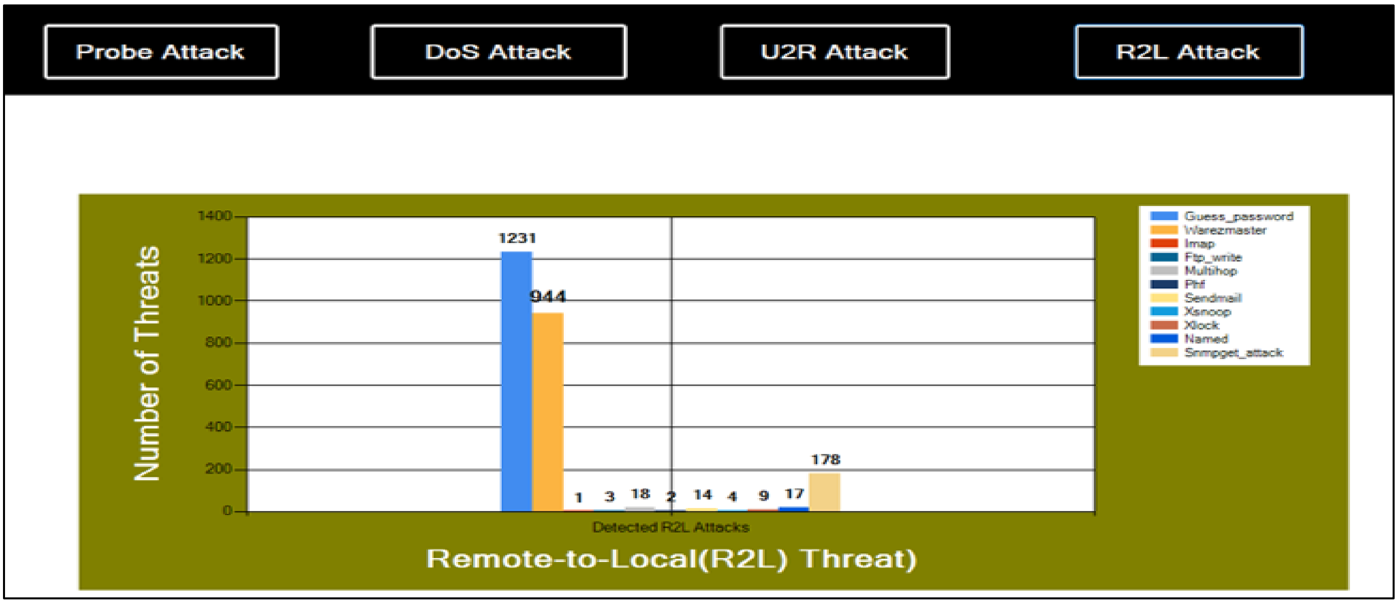


Fig 11 R2L Attack Type Classification Form

Fig 11 depicts detected root-to-local attacks, which include sending packets to the targeted machine in order to learn about the user's behavior and get access to the system. The guess password recorded the most cases (1231); followed by warezmaster (944), imap, ftp write, multihop, phf, sendmail, xsnoop, xlock, named, and snmpget attacks, with imap having only one case.

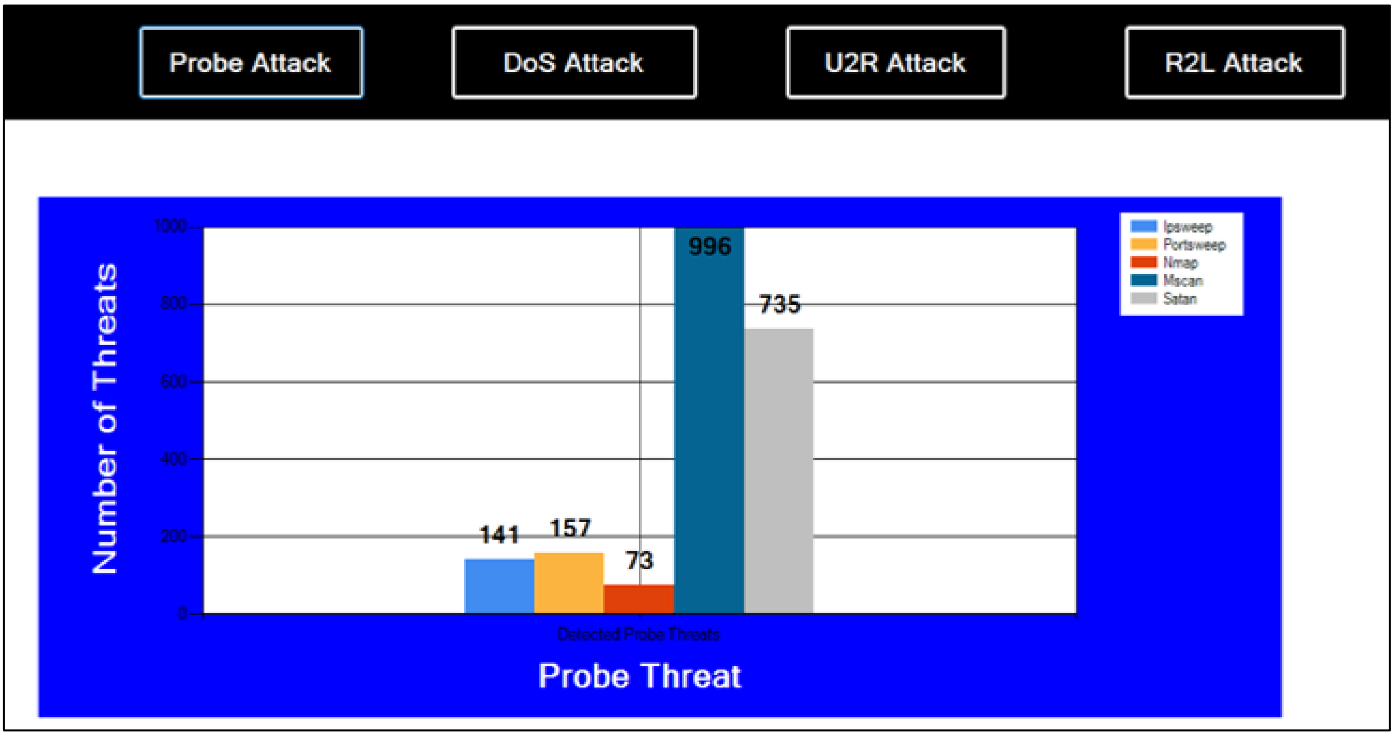


Fig 12 Probe Attack Type Classification Form

Fig 12 shows the chart generated when a user selects the Probe attack icon.The "Probe attack" icon displays a chart of ipsweep, portsweep, mscan, nmap, and satan cases found in the suggested system dataset. The mscan probe attack had the highest score of 996, followed by satan and nmap, respectively.

V. CONCLUSION

The existing logistic regression (LR) technique in operation has proven to be highly ineffective when dealing with IDS for local and wide area networks to maintain security issues, as opposed to the proposed transfer learning (TL) method, which is effective, reliable, and accurate. According to the study above, the proposed system has solved the major errors of the current system. We therefore, conclude that the proposed system technique is promising in terms of detecting Normal cases, DoS, R2L, U2R, and Probe IDS attacks which is much better than the existing method (system) in terms of accuracy and error rates,

REFERENCES

[1]. Albulayhi, K., Abu Al-Haija, D., Alsuhibany, S. A.,. Jillepalli, A. A.,. Ashrafuzzaman,M. and Sheldon, F. T.(2025) IoT intrusion detection using machine learning with a novel high performing feature selection method, Applied Sciences, 12(10), 1-20, doi: 10.3390/app12105015.

[2]. Ambusaidi M. et al (2022), “Building an intrusion detection system using a filter-based feature selection algorithm”, International Journal of Innovative Research & Studies: 112(23):25-27.

[3]. Butun Ismail(2023)., “Prevention and Detection of Intrusions in Wireless Sensor Networks”, Scholar Commons: 15

[4]. Chawla S. (2023), Deep Learning based Intrusion Detection System for Internet of Things: 2

[5]. Diogenes, Y. and Ozkaya,E. (2023) Counter modern threats and employ state-of-the-art tools and techniques to protect your organization against cybercriminals, in Cybersecurity – Attack and Defense Strategies, 2nd Editio., Packt Publishing Ltd., 2-30

[6]. Furnell, S., Jusoh, A. and Katsabas, D. (2024), The Challenges of Understanding and Using Security: A Survey of End-users, Computer and Security, 25, 27–35.

[7]. Ghosal A. and Halder S. (2022), Intrusion Detection in Wireless Sensor Networks: Issues, Challenges and Approaches, Research Gate: 1(12), 2-23.

[8]. Hu Y. et al (2020), ” A survey of intrusion detection on industrial control systems”, International Journal of Distributed Sensor Network: 2-3

[9]. Mohammad M. et al (2021), A Novel Local Network Intrusion Detection System Based on Support Vector Machine”, Journal of Computer Science, 7(10), 1560-1568.

[10]. Parag K. et al (2021), “Intrusion Detection System for Cloud Computing”, International Journal of Scientific & Technology Research,1(4), 12-16.

[11]. Sheikh A. et al (2022), Analytical Study on Hybrid Approach towards Intrusion Detection System for Wireless Sensor Network: 3

[12]. Singh, A., Amutha, J. Nagar, J., Sharma, S. and C.-C. Lee, (2022) LT-FS-ID: Log-transformed feature learning and feature-scaling-based machine learning algorithms to predict the k-barriers for intrusion detection using wireless sensor network, Sensors, 22(3), 1070, doi: 10.3390/s22031070.

[13]. Solanki, S., Gupta, C. and Rai, K.(2020), A Survey on Machine Learning based Intrusion Detection System on NSLKDD Dataset, International Journal of Computer Applications, 176(20), 121- 128.

[14]. Stosic L. (2022), “Computer Security and security technmologies”, Research Gate: 25.

[15]. Surantha N. and Wicaksono W. (2019), An IoT based House Intruder Detection and Alert System using Histogram of Oriented Gradients”, Journal of Computer Science, 31(41), 1101-1121

[16]. Subramaniam M and Kathirvel K (2020), Improved Intrusion Detection and Response System for Wireless Sensor Network, International Journal of Forensic Science 10(2), 23-30.

[17]. Talukder, A., Hasan, K. F., Islam, M., Uddin, A., Akhter, A., Yousuf, M., Alharbi, F., and Moni, M a.(2023), A Dependable Hybrid Machine Learning Model for Network Intrusion Detection, Journal of Information Security and Applications, arXiv:2212.04546v2 [cs.CR] 27 Jan 2023, 1-44.

[18]. Tan, Z. Jamdagni, X. He, X. Nanda, P. and Liu, R. P. (2024), A system for denial-of-service attack detection based on multivariate correlation analysis, IEEE Transactions on Parallel and Distributed Systems, 25, 447–456.

[19]. Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., and Liu, C. (2021). A survey on deep transfer learning. In Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27 (pp. 270-279).

[20]. Tsiakmaki, M., Kostopoulos, G., Kotsiantis, S., & Ragos, O. (2020). Transfer learning from deep neural networks for predicting student performance. Applied Sciences, 10(6), 2145.

[21]. Vadhil, F. A., Salihi, M. L. and Nanne, M. F.(2024), Machine learning-based intrusion detection system for detecting web attacks, IAES International Journal of Artificial Intelligence (IJ-AI), 13(1), 711~721

[22]. Usha D. And Suganthi S. (2023), A Survey of Intrusion Detection System in IoT Devices, International Journal of Advanced Research (IJAR): 23, 12-23.

[23]. Wang K. and Stolfo S., (2019), “Anomalous Payload-based Network Intrusion Detection”:12-23

[24]. Wazid M. (2021), Design and Analysis of Intrusion Detection Protocols for Hierarchical Wireless Sensor Networks, Design and Analysis of Intrusion Detection Protocols for Hierarchical Wireless Sensor Networks, 1(23), 23-28.

[25]. Yao J. (2020), An Enhanced Support Vector Machine Model for Intrusion Detection, 4(5),1-4.