JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

ANOMALY DETECTION IN NETWORK TRAFFIC USING DEEP LEARNING TECHNIQUES

[1]Likith Gowda S, [2]Rini Sharma, [3]Dr. Jagruthi H

[1]Under Graduate Student, [2] Under Graduate Student, [3]Associate Professor
[1]Information Science and Engineering,
[1]BNM Institute of Technology, Bengaluru, India

Abstract : The frequency of cyberattacks has increased due to the Internet's rapid proliferation. Intrusion detection systems (IDS) are being used to protect system security. IDS is still experiencing some issues making its categorization function better. First off, the complexity of high-dimensional qualities presents a barrier to the effectiveness and speed of the categorization for IDS. Second, the classification performance of the traditional Stacking approach is directly influenced by the basic classifiers. To solve the two difficulties described above, we offer a hybrid intrusion detection system based on a weighted Stacking classification method and a CFS-DE feature selection strategy. To minimize the dimension of the features, we employed the CFS-DE algorithm, which looks for the ideal feature subset. Then, a weighted Stacking technique is recommended to improve classification performance by decreasing the weights of the fundamental classifiers with bad training results and increasing the weights of those with positive outcomes. The model thus enhances classification efficiency and accuracy. For all experiments in this work, the NSL-KDD and CSE-CIC-IDS2018 data sets were utilized.

IndexTerms **- Intrusion Detection System(IDS), Deep Learning, Machine Learning.**

I. INTRODUCTION

Network attacks are one of the causes of the strange phenomena observed in the operation of the network hardware and the transfer of traffic across the network. Network traffic anomalies may cause a denial of service in this network's equipment by causing a single channel to operate incorrectly or even entire network segments. Because attackers employ unique strategies, network attacks are always evolving. Modifications to the hardware and software also have an impact. The most crucial defense mechanisms against the complex and expanding network threats are intrusion detection systems (IDSs) and intrusion prevention systems (IPSs). Performance evolutions for anomaly-based intrusion detection techniques are inconsistent and inaccurate since there aren't enough trustworthy test and validation datasets.

The majority of the eleven datasets that have been made public since 1998 are out-of-date and unreliable, according to our studies of them. Some of these statistics do not accurately reflect current trends due to their lack of diversity and volume, their underrepresentation of the variety of known assaults, and their anonymization of packet payload data. Some also lack information and feature sets. The CICIDS2018 dataset includes recent, safe assaults that closely resemble PCAPs from the real world. It also includes the results from the CICFlowMeter network traffic analysis, which is organized into flows according to the time stamp, source and destination IP addresses, source and destination ports, protocols, and attack (CSV files). There is also a definition of extracted characteristics provided.

Building this dataset with realistic background traffic generation as our top priority. We profiled the abstract behavior of human interactions using our suggested B-Profile system (Sharfuddin, et al. 2016), which also creates naturalistic benign background traffic. For this dataset, we created the abstract behavior of 25 users based on the HTTP, HTTPS, FTP, SSH, and email protocols. One of the sources of the odd phenomena seen in the operation. of the network hardware and the transmission of traffic over the network is network attacks. By forcing a single channel to operate erroneously or even entire network segments, network traffic anomalies may result in a denial of service in this network equipment. Because attackers employ unique strategies, network attacks are always evolving. Modifications to the hardware and software also have an impact. This study involves finding unusual network traffic patterns. Law enforcement organizations must find ways to simplify investigations and help bring offenders to justice in order to keep up with the rising crime rate in today's society. Utilizing face recognition technology to identify and confirm the culprit is one such method. Finding patterns in data that do not match expected behavior is known as anomaly detection. In this study, we're looking for unusual network traffic patterns. The detection accuracy rating ranges from 90% to 95%. Now, we wish to increase the accuracy value for identifying unusual network activity.

II. RESEARCH METHODOLOGY

The process of identifying anomalies in network traffic has been studied extensively. An algorithm can be used to foresee network abnormalities using the data that is already available, such as packet size, time length, protocol type, and other aspects of network traffic. Additionally, clustering algorithms and fuzzy inference systems can be utilized with well-known machine learning techniques including logistic regression, support vector machines, gradient boosting, neural networks, k-nearest neighbor, and random forest. Anomalies in network traffic are defined as deviations from normal patterns and behavior that might be a sign of malicious or unauthorized activity. The objective is to identify and prevent such irregularities before they harm the network or the systems connected to it.

Data Collection

Data on network traffic is gathered in this step from a variety of sources, including network sensors and traffic logs. A data processing system is then used to load the data for additional analysis.

Data Preprocessing

The collected data is usually noisy and contains irrelevant information, before it can be utilized to train the model, it must first be preprocessed. For example, addressing missing values, scaling the data, and encoding categorical variables are activities involved in this.

Feature Extraction

Since network traffic data is frequently multidimensional and complicated, it's critical to isolate pertinent elements from the data that may be fed into the model. To find patterns in the data, this could entail using statistical methods or machine learning algorithms.

Model Selection

Given the complexity of deep learning models, it is crucial to select the right architecture and hyperparameters for the task at hand. Convolutional neural networks, recurrent neural networks, and transformer models might be tested, and various hyperparameters like learning rate, batch size, and number of layers may also need to be adjusted.

Model Training

Once a model architecture and hyperparameters have been selected, the model can be trained on the preprocessed data using an appropriate optimization algorithm, such as stochastic gradient descent or Adam. During training, the model gains the ability to distinguish between patterns in the data that are indicative of typical network traffic and patterns that are suggestive of aberrant behavior.

Model Evaluation

After the model has been trained, it is crucial to assess how well it performs using a different set of test or validation data. Various criteria, including accuracy, precision, recall, and F1 score, can be used to accomplish this. It may also be useful to visualize the model's predictions and decision boundaries. It is possible to assess the trained model's performance to see whether it has successfully learnt to differentiate between regular and anomalous network data. If not, any necessary corrections may then be made to increase the model's accuracy and efficacy.

Model Optimization

It could be necessary to further optimize the basic model after evaluation in order to boost performance. This can entail changing the model architecture or tweaking the hyperparameters.

Deployment and Prediction

The trained model can be used to identify anomalies in network traffic data once it has been evaluated and found to be satisfactory. This entails using the model to forecast abnormal events in real-time based on the input data and incorporating it into the production environment.

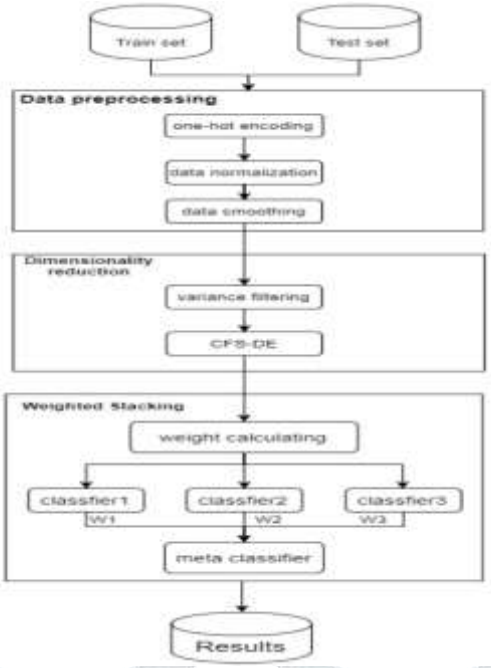

Figure 1: System Architecture

III. MODELING AND ANALYSIS

The Data Flow Diagram (DFD), also known as a bubble chart, is a graphical formalism used to represent a system. The diagram depicts the input data to the system, the various processing carried out on this data, and the output data generated by the system. The diagram uses a series of bubbles or nodes to represent the processes and data stores, and arrows to represent the information flow between them.

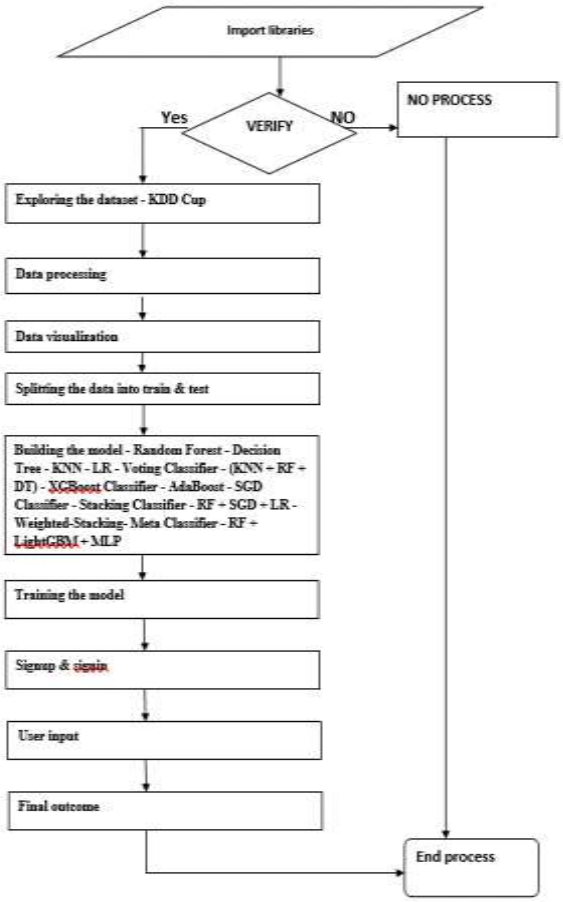

Figure 2: Dataflow Diagram

A sequence diagram shows how various system items interact with one another. The fact that a sequence diagram is time-ordered is crucial. This indicates that the precise order of the objects' interactions is displayed step by step. The sequence diagram shows how various things communicate with one another by sending "messages".

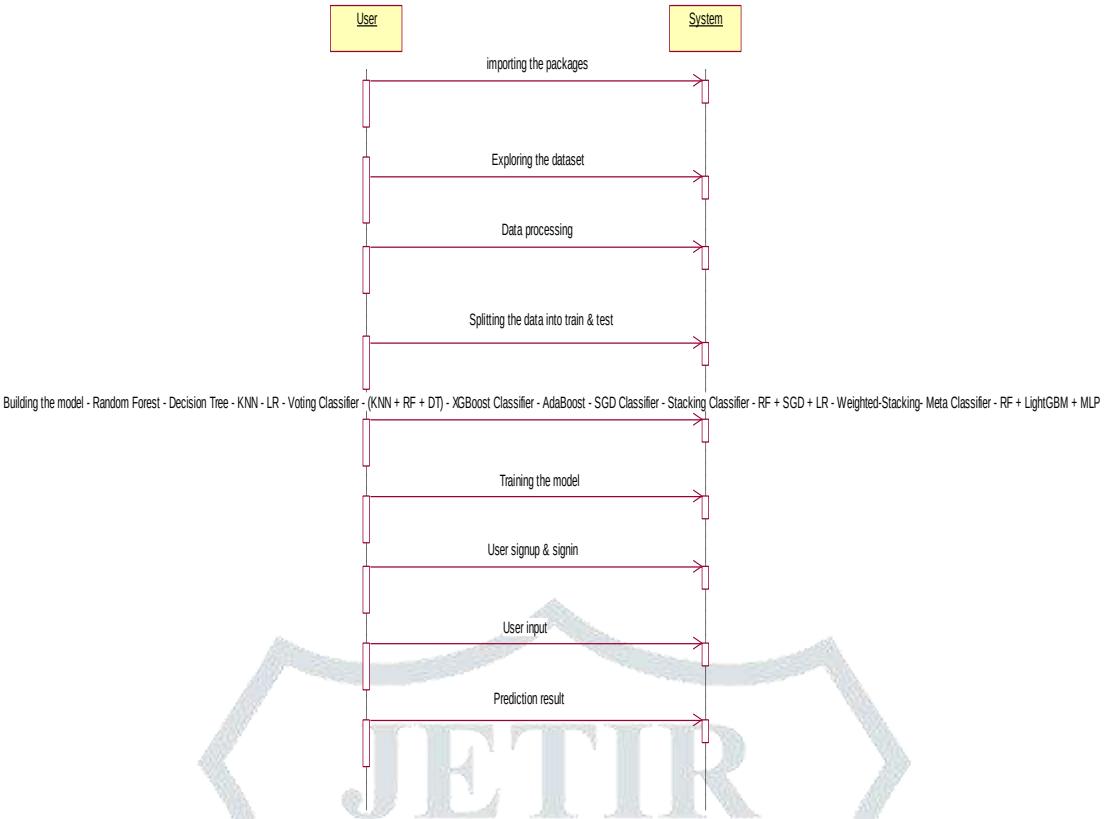

Figure 3: Sequence Diagram

Random Forest

An incredibly well-liked supervised machine learning approach called the Random Forest approach is utilised to solve classification and regression issues. We know that a forest comprises numerous trees, and the more trees more it will be robust.

Decision Tree

The non-parametric supervised learning approach used for classification and regression applications is the decision tree. It is organized hierarchically and has a root node, branches, internal nodes, and leaf nodes.

KNN

The k-nearest neighbors algorithm, sometimes referred to as KNN or k-NN, is a supervised learning classifier that employs proximity to produce classifications or predictions about the grouping of a single data point.

Weighted-Stacking- Meta Classifier - RF + LightGBM + MLP- LightGBM

is a quick, distributed, high-performance gradient boosting framework built on decision tree techniques that may be utilised for many different machines learning tasks, including classification and ranking. Kagglers have used it to succeed in data science competitions

.

IV. RESULTS AND DISCUSSION

The project worked as expected. The proposed architecture had the maximum accuracy and an acceptable value of loss metric among all the tested architectures. For the purpose of such validation tensor board support “Accuracy” metrics. This metrics allows to see the loss, validation loss, accuracy and validation accuracy over every epoch during the training phase of the neural network.

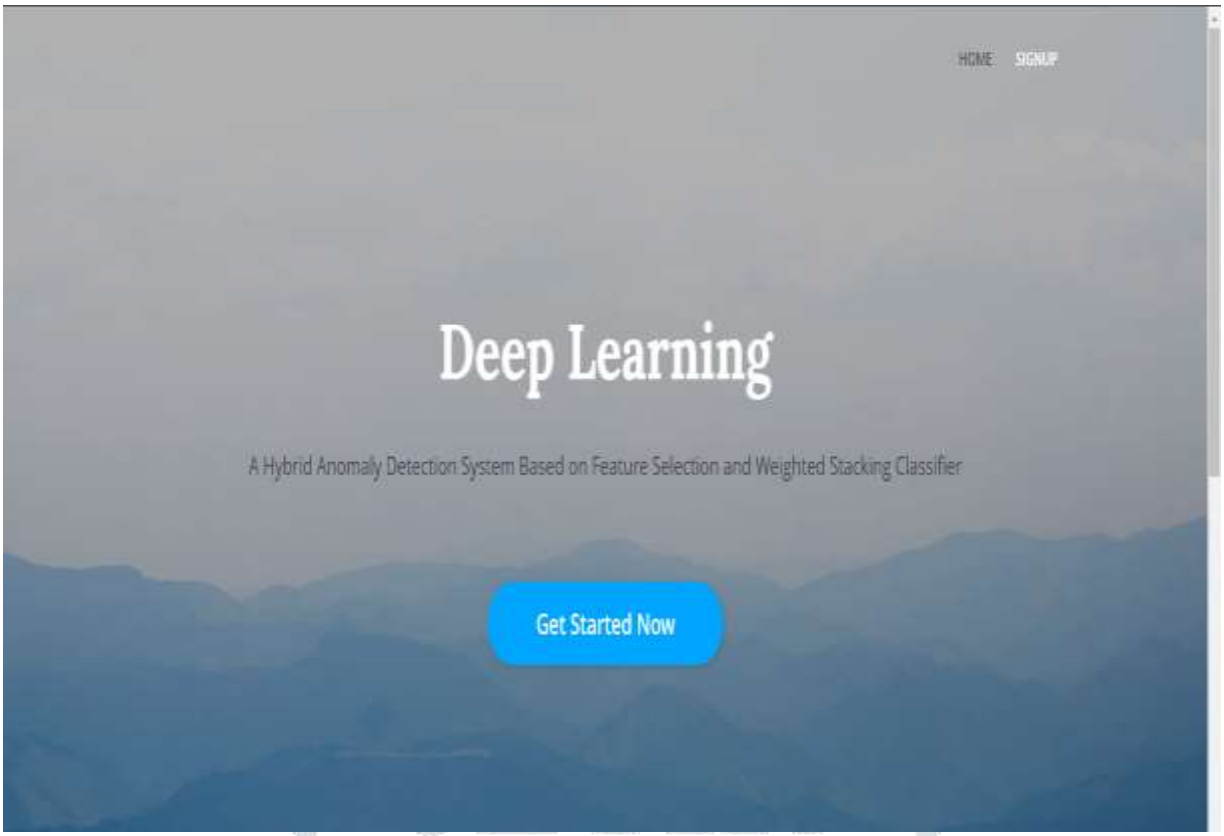

Figure 4: Home Page of the website

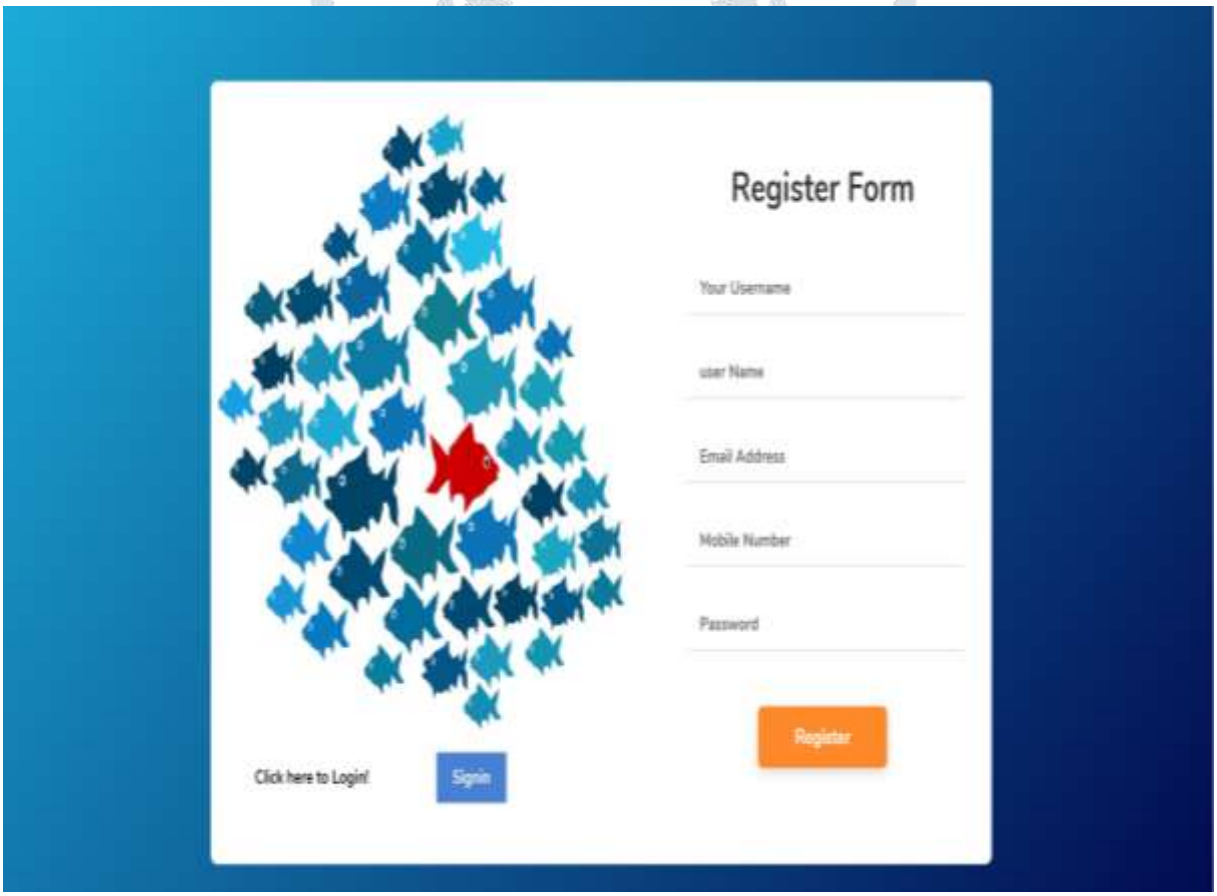

Figure 5: Signup Page
Figure displays the page to signup

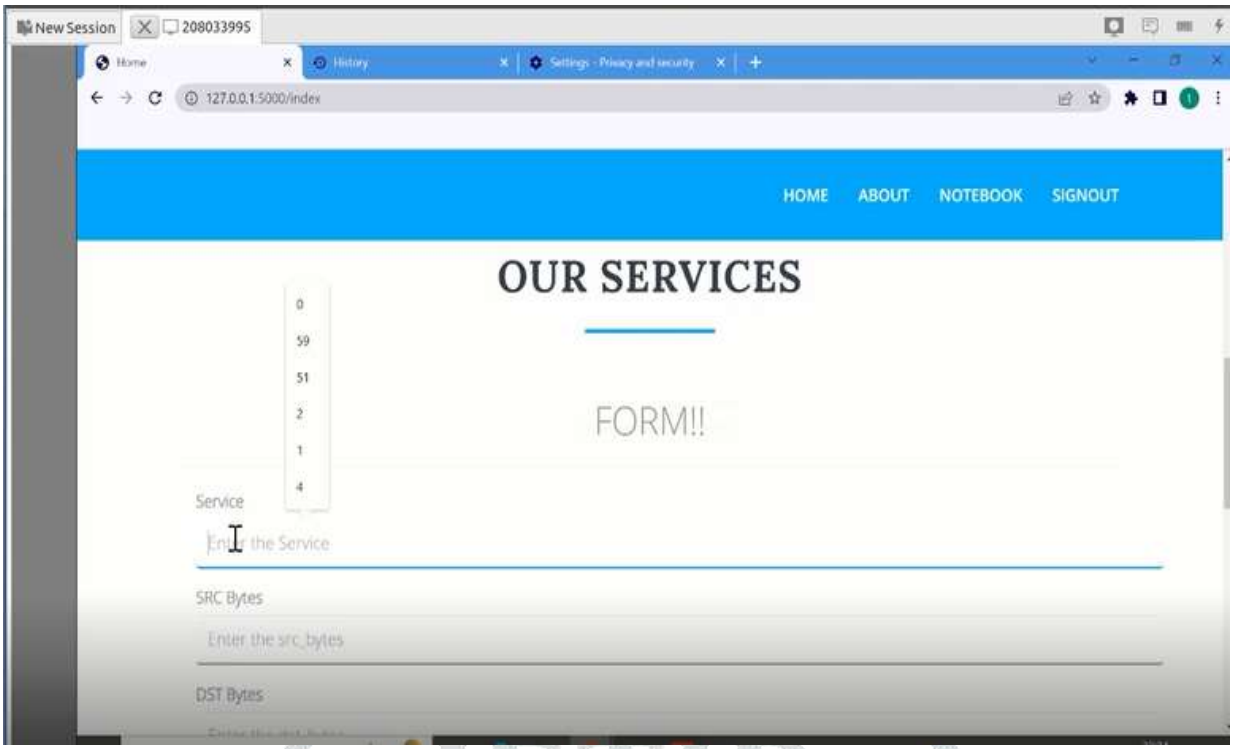

Figure 6: Prediction Page
Figure displays where the prediction happens

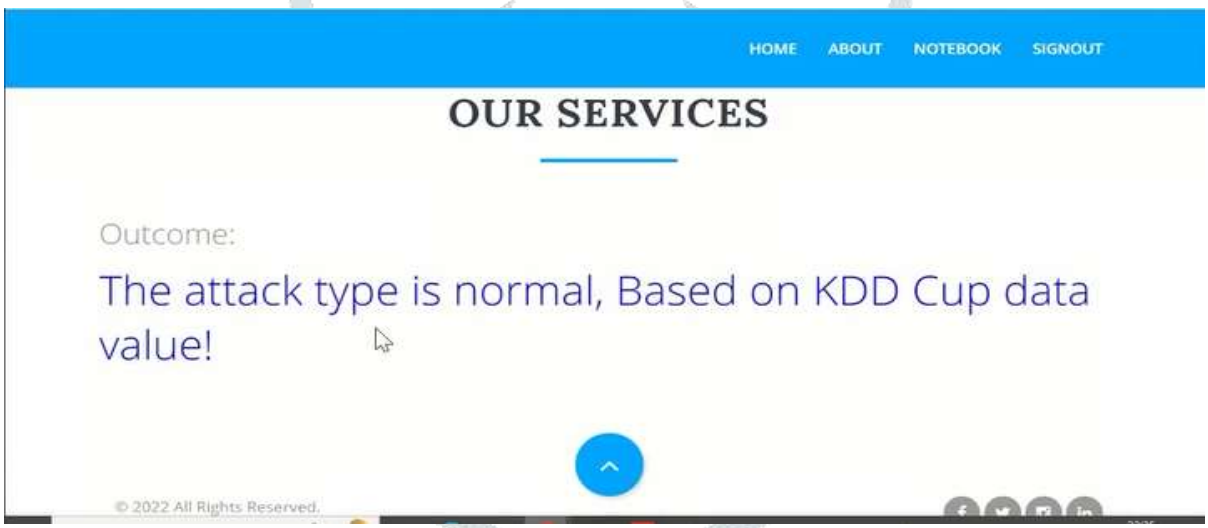

Figure 7: Final Results

V. CONCLUSION

The significance of intrusion detection has grown as Internet technologies have advanced. We provide a hybrid framework that combines the weighted Stacking classification method and the CFS-DE feature selection technique to reduce high dimensional data and improve classification performance even more.

We initially recommend applying the CFS-DE feature selection method to identify the ideal feature subset. To further enhance IDS's classification performance, the weighted Stacking classification algorithm is then introduced. In this study, Random Forest, XGBoost, and KNN served as the base classifiers, with Logistic Regression serving as the meta classifier.

Finally, we test our proposed IDS using the NSL-KDD and CIC-IDS2018 data sets. For the CSE-CIC-IDS2018 with a subset of 20 characteristics, our model's accuracy rate, recall rate, and F1-score are 87.44%, 89.09%, and 88.25%, respectively. It is impressive that our model, when applied to the subset of 15 features from the KDDTest data set, gets the greatest accuracy rate of 96.87%, precision rate of 96.88%, recall rate of 95.87%, and F1 score of 94.88%. Our suggested CFS-DE-weighted-Stacking IDS, however, reduces the duration from 776.74s to 144.50s on CSE-CIC-IDS2018 and from 299.90s to 168.93s on KDDTest when compared with the all-features data set.

VI. REFERENCES

[1] Imamverdiyev, Y., & Sukhostat, L. (2016, October). Anomaly1detection in network1traffic using extreme learning machine. In 20161IEEE 10th international conference1on application of information1and communication1technologies (AICT) (pp. 1-4). IEEE.

[2] Ali, W. A., Manasa, K. N., Bendechache, M., Fadhel Aljunaid, M., & Sandhya, P. (2020). A review1of current machine learning approaches1for anomaly detection in network traffic. Journal of Telecommunications and the Digital Economy, 8(4), 64-95.

[3] Loganathan, G., Samarabandu, J., & Wang, X. (2018, May). Sequence to sequence1pattern learning algorithm for real-time anomaly detection1in network traffic. In 2018 IEEE Canadian1Conference on1Electrical & Computer Engineering1(CCECE) (pp. 1-4). IEEE.

[4] Nawir, M., Amir, A., Yaakob, N., & Lynn, O. B. (2019). Effective and efficient1network anomaly detection system using1machine learning algorithm. Bulletin of Electrical1Engineering and Informatics, 8(1), 46-51.

[5] Sharma, Y. K., & Rokade, M. D. (2019). Deep and machine learning approaches for1anomaly-based intrusion detection of imbalanced network traffic. IOSR1Journal of Engineering, 63-67.

[6] Radford, B. J., Apolonio, L. M., Trias, A. J., & Simpson, J. A. (2018). Network traffic1anomaly detection1using recurrent neural networks. arXiv preprint arXiv:1803.10769.

[7] Min, B., Yoo, J., Kim, S., Shin, D., & Shin, D. (2021). Network anomaly1detection using memory-augmented deep autoencoder. IEEE Access, 9, 104695-104706.

[8] Afza, A. A., & Uddin, M. S. (2014, March). Intrusion detection1learning algorithm through network mining. In 16th Int'l Conf. Computer1and Information)Technology (pp. 490-495). IEEE.