Adaptive Detection of Network Attacks Using a Hybrid Transformer Architecture

Mohammed Abdul Rasheed
M.Tech Cyber Security Student, JNTUH University College of Engineering Jagtial, Telangana, India

Suresh Kumar Sanampudi
Associate Professor of Information Technology, JNTUH University College of Engineering Jagtial, Telangana, India - 505501

***Abstract*— The project's primary objectives are to secure cyberspace in the era of the fast evolving Internet technology and to address the escalating issue of attacks in the Internet. The main objective is to improve IDS by going beyond conventional rule-based and keyword-based techniques. To develop a more reliable and flexible approach for detecting network intrusion events, the project looks into the incorporation of AI, in the specific of DL techniques. A more efficient and adaptable intrusion detection system is desperately needed in light of the shortcomings of traditional detection techniques. Due to the growing number of different attack techniques, the intelligent system should be capable of adapting to new attacks on its own without the need to manually formulate rules. The goal of this initiative is to help people and businesses that depend on network services. Enhanced intrusion detection, safeguarding vital data and systems, ensures higher cybersecurity. The implications of the work in the project may have a significant impact on the overall network security area. This proposed model is the Transformer (autoencoder) LSTM model that uses DL. By using the LSTM model, the sequential connections between network traffic can be better understood, making it possible to more accurately detect intrusion events. Various attention processes are adopted in order to select and extract features. Moreover, we are comparing the model with Transformer (autoencoder), DNN and LSTM. To enhance the performance, the Transformer (autoencoder) LSTM model was extended to CNN and CNN+LSTM. The aim of these models is to make even better feature extraction and to exploit spatial correlation from network data.**

Keywords*— *intrusion detection, transformer, LSTM, deep learning.

I. INTRODUCTION

As Internet technology advances, people's lives and careers are increasingly impacted by the Internet. Therefore, this problem of cyber security has also acquired greater importance. For example, there are attacks on the web, which are among the most important network security issues in recent years [1]. IDS are extensively researched in order to guarantee security.

Traditional methods for network attack detection are rule-based or keyword matching. The characteristics and description rules of the recognised assault patterns are stored in the rule bases. These detection methods are based on regular matching in order to detect attacks. They have significant limitations even tho they are capable of detecting current network attack modes. On the other hand, the intrusion detection technique based on matching keywords or a rule base can only detect the intrusion pattern which has been written in the rule base, it cannot respond to new intrusion pattern. There are some techniques that the hackers can easily circumvent such regulations. Rule base building and maintenance, however, rely on security experts and are costly in terms of maintenance, and therefore not widely adopted. Furthermore, with the increasing diversity of attack methods, traditional methods for intrusion detection are no longer sufficient to ensure network services security. More efficient intrusion detection technique must be suggested. With the development of AI technologies, the research of intrusion detection systems based on ML and DL techniques is attracting attention in recent years.

The ML-based intrusion detection approach is more reliable than the conventional rule-based or keyword-matching approach and it also works well in real world applications. Such an approach, however, has very stringent demands with respect to feature selection. If the features selected are inappropriate, the effectiveness of intrusion detection system will be drastically reduced. The rules of intrusion detection based on DL are more complex. This type of approach can accomplish the independent selection of suitable traffic features in the training process and the extraction of complex features based on preexisting simple features by utilising the end-to-end training process from the original traffic to the classification results. Therefore, the DL based method is more ubiquitous than conventional ML based intrusion detection method [2].

CNNs, RNNs, Transformers, and other traditional DL models are among the various types of neural network models utilised in modern DL. These basic models have been widely utilized in many disciplines. That's why, in this study, an Intrusion Detection model using Transformer & LSTM is proposed [8,9]. In order to identify network intrusion events, the model realises the sequence relationship of traffic context using an LSTM model and employs several attention mechanisms to finish feature selection and extraction. As Internet technology advances, people's lives and careers are increasingly impacted by the Internet. It makes the problem of the security of a cyberspace also more important. For example, web attacks have grown to be one of the foremost problems in network security lately [1]. IDS are extensively researched in order to guarantee security.

Traditional methods to detect network attacks rely on a rule base or keyword matching. The traits and description rules of recognised assault patterns are contained in the rule bases. The detection techniques employ standard matching to detect the attacks. Despite being able to identify live network attack modes, they have major drawbacks. On the other hand, the intrusion detection technique based on keywords similarity or a rule base can only recognise the attack patterns already existed in the rule base, and unable to respond new attack patterns. But with some tricks the hackers can easily circumvent these regulations. But, the building and maintenance of the rule base rely on security experts, leading to costlier maintenance and lack of the widespread adoption. Furthermore, the increasing number of intrusion methods makes traditional intrusion detection methods insufficient to ensure that services on a network are secure. More efficient techniques of intrusion detection should be proposed. With the development of AI technologies, intrusion detection systems have been studied recently based on ML and DL techniques.

The ML-based intrusion detection approach is more reliable than the conventional rule-based or keyword-matching approach, and the approach possesses a good performance in the real world applications. But, this kind of method is rather strict on feature selection. If the features are not selected properly, the intrusion detection system will be greatly affected. Rules for intrusion detection based on DL are

more complex. In this way, the independent selection of appropriate traffic features in the training phase and the extraction of complex features based on simple ones already known can be achieved, by using the end-to-end training from the original traffic to the classification results. Therefore, DL-based method of intrusion detection is more ubiquitous than the conventional ML based approach [2].

There are a number of different types of neural network models that are used in modern DL, such as CNNs, RNNs and Transformers. These basic models have been widely employed in various areas. Therefore, Transformer & LSTM based intrusion detection model is proposed in this work [8,9]. To identify network intrusion events, the model is realized by using LSTM model to find the sequence relationship of traffic context and utilizes several attention mechanisms to complete feature selection and feature extraction.

II. Related Work

In the article "An empirical analysis of cyber security incidents at a large organization", the author describes how security engineers are faced daily with a multitude of cyber security issues. Most issues are solved with common answers, but there are some issues that require orders of magnitude of investigation and solution. How security operation teams in organisations will react to deal with extreme occurrences remains unclear. In this paper, we analyze the statistical properties of sixty thousand security events collected over six years at a large enterprise, and find that the distribution of the consequences of a security incident is typically extremely power-law tailed. Over time, tho, this distribution of incident severity gets less skewed, indicating that the organization under investigation has been able to lessen the impact of significant incidents. We illustrate this result with a case study which concentrates on the empirical impact of full disc encryption on the severity of lost or stolen devices.

The study “Detection and classification of anomaly intrusion using hierarchy clustering and SVM” discusses anomaly detection which is a new area of intrusion detection that has received much attention in recent years and which is useful for detecting unexpected or new attacks. In this paper a novel evolutionary algorithms based combination of SVM and FCM is proposed as a hierarchical model for intrusion detection for anomaly. Intrusions are detected using fuzzy interval and the membership function, and the technique is extended from the previous hard classification to soft classification. To enhance the intrusion detection performance, a fuzzy error correction sub interval is then realised. If the result of detection of a data instance is in this range, the data will be re-detected. The suggested model can successfully identify the great majority of network attack types, according to experimental data, which offers a workable way to address the issues of false alarm rate and detection rate in anomaly intrusion detection models.

[3] Given the pervasiveness of cyberattacks that spread quickly and change over time, the paper "Hybrid Intrusion Detection System Using ML Techniques in Cloud Computing Environments" discusses how intrusion detection is a crucial tool for creating a safe and reliable cloud computing environment. Cloud Computing is a strong enabler for drastically enhancing the cost metrics in our existing working paradigm of consolidations of resources, platforms and services. Almost all cloud networks use Internet to provide the service on which they operate and these networks suffer from several security issues. Hence it is necessary to introduce IDS in the cloud environment to identify all the new and unknown threats in addition to known threats based on the signatures. When there is a notable deviation in one or more underlying system or network activities, we consider a system or network "anomalous" event to be equivalent to a "intrusion" event. The aim of some recently proposed ideas is to build a hybrid detection scheme that brings together the advantages of signature based detection techniques and ability to detect unknown attacks through abnormalities. In order to increase the anomaly detection system's accuracy, we provide a network-based anomaly detection system at the Cloud Hypervisor level that makes use of a hybrid approach, which combines the K-means clustering technique with the SVM classification algorithm. The proposed technique is evaluated on the UNSW-NB15 study dataset, and its results are compared with previous studies. Our proposed K-means clustering model is slightly more accurate than the other models. However, for supervised methods the accuracy of the SVM model is still poor.

This study “Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection” emphasizes on the importance of a good NIDS to safeguard sensitive systems in cyberspace from potential threats. In real networks, there are many different types of network traffic data and they may all be associated with different types of abnormal behaviours and should be fully exploited in the NIDS. In the real network environment, the detection of aberrant behaviour by traditional learning-based methods has a weak ability to satisfy the requirements to the complicated network traffic information. Existing approaches have not taken into account the impact of multiple data modalities, and the mutual support of diverse data aspects. This study suggests a MFFSEM that can successfully identify anomalous behaviours in order to allay the worries. Based on different features of traffic information, including time, space and load, several basic feature sets are generated for the purpose of accurately investigating the connotation of traffic information. Then, a number of large feature sets are combined to meet the requirements of practical aberration behavior detection considering the correlation and association of the basic feature sets. In particular, by combining ensemble learning with multiple large feature sets, an efficient multi-dimensional global anomaly detection model is realized. MFFSEM greatly outperforms the basic and meta classifiers used in our method, according to experimental results on the datasets KDD Cup 99, NSL-KDD, UNSW-NB15, and CIC-IDS2017. It also has a higher detection rate than other commonly used ensemble techniques.

[5] The article "A novel network intrusion detection system based on CNN" discusses how crucial NIDS are to network security. It has the ability to identify malicious communications and stop network intrusions. Traditional methods include ML techniques such as SVM, Bayesian classification, DT and K-means. There are definite limitations in the traditional ML methods, and they require manual feature selection. This paper introduces a revolutionary NIDS system, based on the CNN. We use both extracted features and real network traffic to construct DL-based detection models. Our extensive testing is done using well known benchmark datasets. The results validate the effectiveness of our system and demonstrate that raw traffic based model is more accurate than extracted features based model.

III. Materials And Methods

i) Proposed Work:

In this study, we suggest a novel intrusion detection system that makes use of the Transformer (autoencoder) LSTM model. We combine the Transformer and LSTM-based architectures, and use multiple attention mechanisms to extract and select features. By improving the comprehension of sequential linkages in network traffic, the LSTM model makes it possible to accurately identify intrusion events. The algorithms used are Min-Max scaling, label encoding for pre-processing and feature selection using SelectPercentile and mutual_info_classif. To assess and compare the above

suggested system, it is benchmarked against DNN, LSTM, and Transformer (autoencoder). To enhance the performance, CNN and CNN+LSTM models were added to the Transformer (autoencoder) LSTM model. These models seek to improve feature extraction even more and take advantage of spatial correlations in network data. A Flask framework was also developed using SQLite, to allow signup and signin features for user testing. This system ensures that the improved intrusion detection will be easy to assess and users will be able to input data and receive results.

ii) System Architecture:

The KDDCup dataset input is the first step in the system design, where it goes thru feature selection and data preprocessing. Following this, the data set is split into training and test data sets. In the process of model creation, the following different designs are used: DNN, LSTM, Transformer (autoencoder), CNN, CNN+LSTM, and Transformer (autoencoder). Performance measures are utilized such as F1 score, accuracy, precision, and recall. In order to create a complete and adaptable intrusion detection system, the last phase includes attack classification and detection.

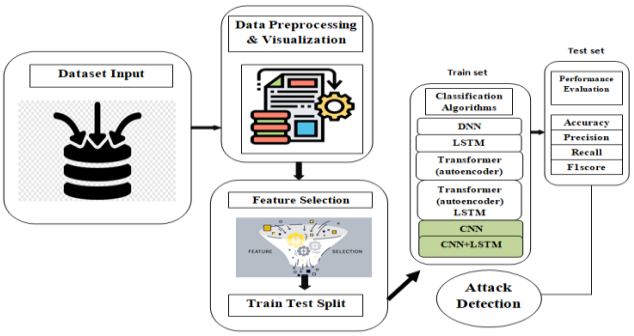

Fig. 1. System Architecture

iii) Dataset Collection

A widely used benchmark data set in the field of intrusion detection research is the KDD Cup data set. The data came from the KDD Cup 1999 competition and provides a representative and diverse sample of network traffic for the evaluation of the effectiveness of intrusion detection systems.

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	...	dst_host_srv_count	dst_host_same_srv_rate	dst_h
0	0	tcp	http	SF	181	5450	0	0	0	0	...	9	1.0	
1	0	tcp	http	SF	239	486	0	0	0	0	...	19	1.0	
2	0	tcp	http	SF	235	1337	0	0	0	0	...	29	1.0	
3	0	tcp	http	SF	219	1337	0	0	0	0	...	39	1.0	
4	0	tcp	http	SF	217	2032	0	0	0	0	...	49	1.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
494016	0	tcp	http	SF	310	1881	0	0	0	0	...	255	1.0	
494017	0	tcp	http	SF	282	2286	0	0	0	0	...	255	1.0	
494018	0	tcp	http	SF	203	1200	0	0	0	0	...	255	1.0	
494019	0	tcp	http	SF	291	1200	0	0	0	0	...	255	1.0	
494020	0	tcp	http	SF	219	1234	0	0	0	0	...	255	1.0	

Fig. 2. Dataset

iv) Data Processing:

Data processing is a way of converting raw data into information that can be used by organisations. Often data scientists are responsible for collecting, organizing, cleaning, verifying, analyzing and representing data in a consumable format such as text or graphs. Data can be processed manually, mechanically and electronically. Increasing the value of information and making decision-making easier are the goals. Businesses can make informed decisions in time and consequently improve their business process. Computer software development and other automated data processing technologies are important in this. It can help to convert huge amounts of data—such as big data—into useful knowledge for decision making and quality control.

v) Feature selection:

The method of selecting most reliable, relevant and non-redundant features to be used in model building is termed as feature selection. With the growing volume and types of datasets, systematic reduction of the size of datasets is significant. Enhancing a predictive model's performance and lowering modeling's computing cost are the primary objectives of feature selection.

Feature selection is one of the major aspects of feature engineering and is the process of selecting the most critical features to be used in the ML algorithms. By removing unnecessary or redundant features and reducing the set of features to those that are most pertinent to the ML model, feature selection approaches are used to lower the number of input variables. The primary advantages of doing feature selection beforehand as opposed to allowing the ML model choose which features are most crucial.

vi) Algorithms:

For complex pattern extraction, a multi-layered neural network architecture, called DNN, is proposed. DNN, made up of input, hidden and output layers, is highly effective at discovering complex relationships in data. DNN is chosen for this project in order to efficiently catch subtle patterns in network traffic. Because of its hierarchical structure, which makes automatic feature extraction easier, it is an effective tool for reliable intrusion detection in the ever-changing and intricate field of cybersecurity.

```
hidden_units = 10     # how many neurons in the hidden layer
activation = 'relu'   # activation function for hidden layer
l2 = 0.01             # regularization - how much we penalize large parameter values
learning_rate = 0.01  # how big our steps are in gradient descent
epochs = 5            # how many epochs to train for
batch_size = 2        # how many samples to use for each gradient descent update
```

```
# create a sequential model
model2 = models.Sequential()

# add the hidden layer
model2.add(layers.Dense(input_dim=14,
                        units=hidden_units,
                        activation=activation))

# add the output layer
model2.add(layers.Dense(input_dim=hidden_units,
                        units=1,
                        activation='sigmoid'))

# define our loss function and optimizer
model2.compile(loss='binary_crossentropy',
               # Adam is a kind of gradient descent
               optimizer=optimizers.Adam(lr=learning_rate),
               metrics=['accuracy'])
```

Fig. 3 DNN

A kind of RNN called LSTM was created to solve the vanishing gradient issue with conventional RNNs. The special architecture of LSTMs, with memory cells and gates, allows them to selectively recall and forget information over long time periods. The choice of LSTM for this study is due to its ability to capture sequential patterns, which is essential for efficient intrusion detection when accurate categorisation relies on the order of events or sequence of links in the traffic.

LSTM

```
from keras.models import Sequential
from keras.layers import Dense, LSTM
from keras.layers import Dropout
from keras import regularizers
import tensorflow as tf

# define a function to build the keras model
def create_model(input_shape):
    # create model
    d = 0.25
    model = Sequential()

    model.add(LSTM(32, input_shape=input_shape, activation='relu', return_sequences=True))
    model.add(Dropout(d))

    model.add(LSTM(64, input_shape=input_shape, activation='relu', return_sequences=True))
    model.add(Dropout(d))

    model.add(LSTM(128, input_shape=input_shape, activation='relu', return_sequences=True))
    model.add(Dropout(d))

    model.add(LSTM(256, input_shape=input_shape, activation='relu', return_sequences=False))
    model.add(Dropout(d))

    model.add(Dense(32,kernel_initializer="uniform",activation='relu'))
    model.add(Dense(1,kernel_initializer="uniform",activation='linear'))
```

Fig 4 LSTM

A neural network architecture called the Transformer (Autoencoder) uses self-attention processes to extract features. The autoencoder architecture of this work can reconstruct the input data and learn robust representations. This self-attention approach allows the model to focus on relevant features, making it suitable for detecting intricate patterns within network traffic data, an essential part of effective intrusion detection.

Fig 5 Transformer (Autoencoder)

The combination of Transformer and LSTM is known as the Transformer (Autoencoder) LSTM. A range of attention mechanisms are used to select and extract features from the model. The sequential knowledge of LSTM improves the ability to capture global relations in data for the Transformer. This combination enhances the model's capacity to capture complex patterns in the network data, making it suitable for the project's diverse and evolving ID needs.

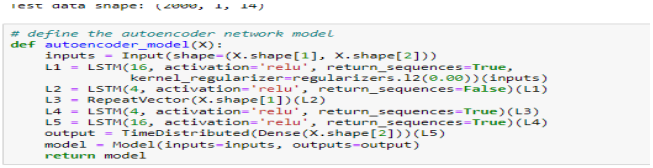

Fig 6 Transformer (Autoencoder) LSTM

DL models called CNNs are made to interpret structured grid data, like pictures. They consist of the convolutional layers which capture the spatial hierarchy via the use of filters over the input data. In this research, CNN was used due to its ability of automatically extracting the relevant features from network traffic data, thus effectively detecting intrusions. Its hierarchical structure and shared weights architecture are well suited to pattern recognition of sequential data, which shares many of the characteristics of network traffic, for accurate intrusion detection.

Fig 7 CNN

CNN and LSTM networks are combined in CNN+LSTM, a hybrid model. LSTM is good at capturing temporal relationships, and CNN is good at extracting spatial features in data. CNN+LSTM was chosen for this project because it can effectively analyse the geographical and sequential relationship between network traffic data, thus enhancing the IDS performance. The CNN+LSTM architecture is well suited for the tasks of capturing patterns which may reveal network intrusions, as it allows for comprehensive feature extraction and learning of sequences.

Fig 8 CNN + LSTM

IV. Experimental Results

Precision: Precision is the percentage of the samples or incidents which are correctly identified as positive. Therefore, the following formula can be used to determine the precision:

Precision is defined as TP/TP + FP (True positives / (True + False positives)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

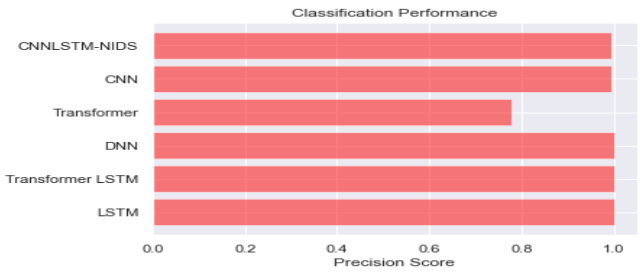

Fig 9 Precision comparison graph

Recall: Recall is a measure of a ML model's ability to identify all relevant examples of a class. It is a measure of the proportion of the model that is successful in capturing instances of a specific class; it is computed as the ratio of correct predictions of positive observations to all the actual positive observations.

$$Recall = \frac{TP}{TP + FN}$$

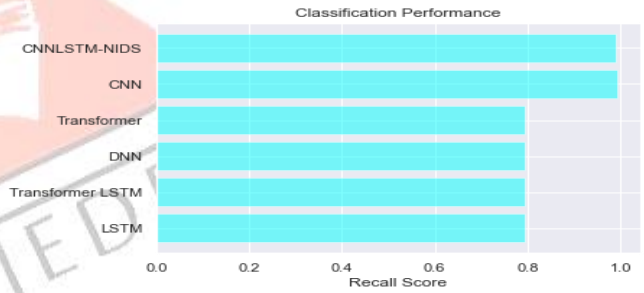

Fig 10 Recall comparison graph

Accuracy: Accuracy is the percentage of correct predictions made on a classification task and is an overall measure of the accuracy of a model's predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

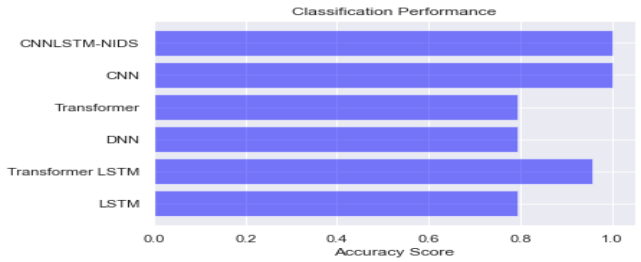

Fig 11 Accuracy graph

F1-Score: F1 score is a ML evaluation measure for the accuracy of a model. Combines a model's recall and precision scores. The accuracy measure is the number of times the model correctly predicted the entire data set.

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

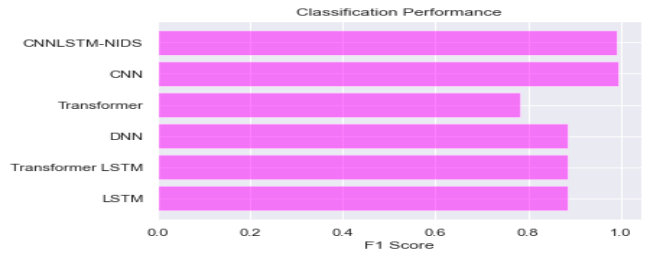

Fig 12 F1Score

	ML Model	Accuracy	Precision	Recall	F1-Score
0	LSTM	0.793	1.000	0.793	0.885
1	Transformer LSTM	0.956	1.000	0.793	0.885
2	DNN	0.793	1.000	0.793	0.885
3	Transformer	0.795	0.778	0.795	0.783
4	**Extension- CNN**	**1.000**	**0.994**	**0.993**	**0.994**
5	**Extension- CNNLSTM-NIDS**	**1.000**	**0.993**	**0.991**	**0.992**

Fig 13 Performance Evaluation

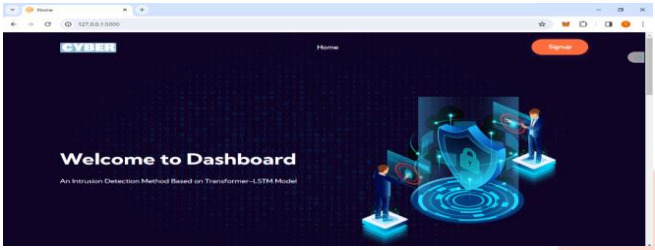

Fig 14 Home page

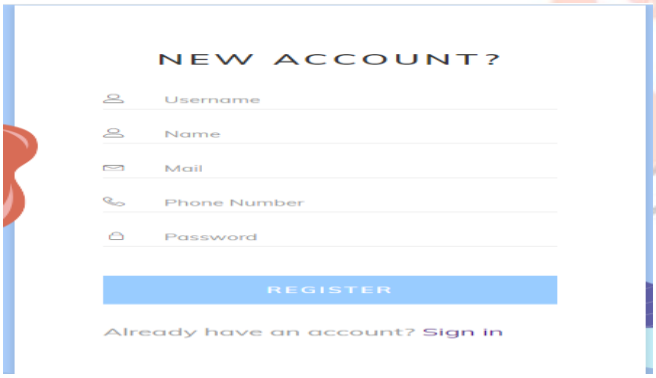

Fig 15 Signin page

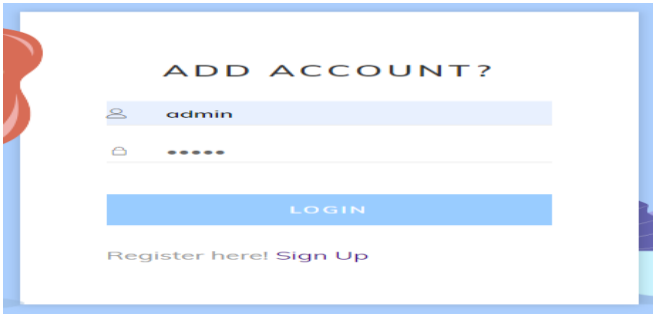

Fig 16 Login page

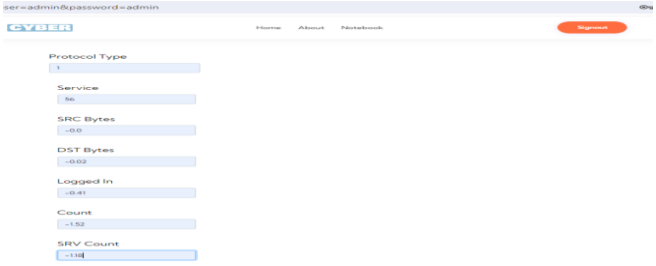

Fig 17 User input

Fig 18 Predict result for given input

V. CONCLUSION

By using careful data processing using Pandas and Keras to optimise the KDD Cup dataset and improve the overall security posture, this study pioneers improved intrusion detection thru DL. A comprehensive and flexible system can be achieved by thoroughly exploring the advantages of DNN, LSTM, Transformer (Autoencoder), and hybrid model (Transformer(Autoencoder) LSTM) for use in intrusion detection. The accuracy is improved by attaching CNN and CNN+LSTM as extensions, which enhances the reliability of the intrusion detection system. The extra models enhance the spatial relationship analysis and feature extraction, which leads to better performance. A user-friendly interface is produced by integrating Flask with SQLite, which promotes more accessibility. To ensure high usability and a wider acceptance, emphasis is laid on user testing, input validation and seamless model predictions. The adaptive systems and real-time features offer a proactive threat response to enhance cybersecurity. Cybersecurity analysts, businesses, and people who depend on network services will all greatly benefit from this endeavour. The system's adaptable design and easy-to-use interface ensure improved intrusion detection for enhanced cybersecurity in a variety of scenarios.

V. FUTURE SCOPE

Future research could involve incorporating adversarial training methods to enhance the model's resistance to sophisticated attacks and ensure the intrusion detection system is more robust. exploring real-time deployment options, improving the model for real-time network traffic analysis, and facilitating timely threat detection and response. To make the model adaptive to new cyberthreats without human interaction, dynamic feature selection algorithms should be designed that dynamically selects the relevant features based on patterns of changing attacks. The intrusion detection system will be more transparent and intelligible to cybersecurity analysts by focusing on improving the interpretability of the model's results and employing techniques such as XAI and attention mechanisms.

REFERENCES

[1] Kuypers, M. A., Maillart, T., & Paté-Cornell, E. (2016). An empirical analysis of cyber security incidents at a large organization. Department of Management Science and Engineering, Stanford University, School of Information, 30. https://fsi-live.s3.us-west-1.amazonaws.com/s3fspublic/kuypersweis_v7.pdf

[2] Tang, C., Xiang, Y., Wang, Y., Qian, J., & Qiang, B. (2016). Detection and classification of anomaly intrusion using hierarchy clustering and SVM. Security and Communication Networks, 9(16), 3401-3411. https://doi.org/10.1002/sec.1547

[3] Mohamad Tahir, H., Hasan, W., Md Said, A., Zakaria, N. H., Katuk, N., Kabir, N. F., ... & Yahya, N. I. (2015). Hybrid machine learning technique for intrusion detection system. https://repo.uum.edu.my/id/eprint/15600

[4] Zhang, H., Li, J. L., Liu, X. M., & Dong, C. (2021). Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection. Future Generation Computer Systems, 122, 130-143. https://doi.org/10.1016/j.future.2021.03.024

[5] Chen, L., Kuang, X., Xu, A., Suo, S., & Yang, Y. (2020, December). A novel network intrusion detection system based on CNN. In 2020 eighth international conference on advanced cloud and big data (CBD) (pp. 243- 247). IEEE. https://doi.org/10.1109/CBD51900.2020.00051

[6] Mirsky, Y., Doitshman, T., Elovici, Y., & Shabtai, A. (2018). Kitsune: an ensemble of autoencoders for online network intrusion detection.

arXiv preprint arXiv:1802.09089. https://doi.org/10.48550/arXiv.1802.09089

[7] Imrana, Y., Xiang, Y., Ali, L., & Abdul-Rauf, Z. (2021). A bidirectional LSTM deep learning approach for intrusion detection. Expert Systems with Applications, 185, 115524. https://doi.org/10.1016/j.eswa.2021.115524

[8] Yang, Y. G., Fu, H. M., Gao, S., Zhou, Y. H., & Shi, W. M. (2022). Intrusion detection: A model based on the improved vision transformer. Transactions on Emerging Telecommunications Technologies, 33(9), e4522. https://doi.org/10.1002/ett.4522

[9] Zhang, Z., & Wang, L. (2022, March). An Efficient Intrusion Detection Model Based on Convolutional Neural Network and Transformer. In 2021 Ninth International Conference on Advanced Cloud and Big Data (CBD) (pp. 248-254). IEEE. https://doi.org/10.1109/CBD54617.2021.00050

[10] Paulauskas, N., & Auskalnis, J. (2017, April). Analysis of data preprocessing influence on intrusion detection using NSL-KDD dataset. In 2017 open conference of electrical, electronic and information sciences (eStream) (pp. 1-5). IEEE. https://doi.org/10.1109/eStream.2017.7950325

[11] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053 c1c4a845aa-Paper.pdf

[12] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780. https://doi.org/10.1109/CISDA.2009.5356528

[13] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009, July). A detailed analysis of the KDD CUP 99 data set. In 2009 IEEE symposium on computational intelligence for security and defense applications (pp. 1-6). IEEE. https://doi.org/10.1109/10.1109/CISDA.2009.5356528