Nanotechnology Perceptions
ISSN 1660-6795
www.nano-*ntp.com*

Multi Stage Attack Detection Using Sequence To Sequence Model

Radha Mogatala[1], Renuka Kondabala[2], V V Nagendra Kumar[3]

[1,2] Assistant Professors, Department Of Information Technology, Vallurupalli Nageswara Rao Vignana Jyothi Institute Of Engineering And Technology, Bachupally, Hyderabad, Telangana, India.
[3] Assistant Professors, Department Of Mca, Rajeev Gandhi Memorial College Of Engineering And Technology, Nandyal, Andhra Pradesh, India.
**Corresponding Author Email Address: Radha_M@Vnrvjiet.In*

September 3rd 2024, submitted
November 10, 2024, accepted
December 4th 2024, published

Multi-stage attack is a kind of sophisticated intrusion strategy that has been widely used for penetrating the well protected network infrastructures. To detect such attacks, state-of-the art research advocates the use of hidden markov model (HMM). However, despite the HMM can model the relationships and dependencies among different alerts and stages for detection, they cannot handle well the stage dependencies buried in a longer sequence of alerts. In this paper, we tackle the challenge of the stages' long-term dependency and propose a new detection solution using a sequence-to-sequence (seq2seq) model. The basic idea is to encode a sequence of alerts (i.e., detector's observation) into a latent feature vector using a long-short term memory (LSTM) network and then decode this vector to a sequence of predicted attacking stages with another LSTM. By the encoder-decoder collaboration, we can decouple the local constraint between the observed alerts and the potential attacking stages, and thus able to take the full knowledge of all the alerts for the detection of stages in a sequence basis. By the LSTM, we can learn to "forget" irrelevant alerts and thereby have more opportunities to "remember" the long-term dependency between different stages for our sequence detection. To evaluate our model's effectiveness, we have conducted extensive experiments using four public datasets, all of which include simulated or re-constructed samples of real-world multi-stage attacks in controlled testbeds. Our results have successfully confirmed the better detection performance of our model compared with the previous HMM solutions.

I. INTRODUCTION

These days, the Internet grows to a great extent by combining the operation techniques with the new information techniques and all of those connected by forming the cyberspace , forming a new Internet of things paradigm . With this trend, a large number of industrial control systems, such as SCADA, PLC etc. previously running on an traditional operating

system and local industrial environment are now connecting to the Internet for a more helpful activity and effective correspondence.

As long as these ICSs are usually served on the critical infrastructures of our national economy and safety. It is necessary to take the network security as an outskirts essential when they open to the public Internet . As a result, the network protections have been widely deployed in the entrance from the Internet to these ICSs, hence largely limiting the intrusion risks to the underlying infrastructures.

However, even the network is well protected by intrusion detection systems (IDS) or firewalls, they are still attracting the hackers or the hacking organizations. Particularly for the case of Internet battle at the national level,the enemy country's ultimate target is usually on the rival's critical infrastructures that are definitely under strict protections . To bypass these protections, attackers are evolving to be more intelligent to discover the protection's weakness and design flaws and sometimes exploit social engineering tricks for penetration, which results a single intrusion task with multiple attacking stages, namely multi-stage attacks .

For example, Havex attacks reported by ICS-CERT 2014 can be considered a type of multi-stage attack. In particular, Havex completes its entry in at least four stages: first, it takes an investigation to find the ICS vendor site that is less secure and toxic to the backdoor on the site's softwares. Second, bypass the ICS protection intended by ICS by downloading malicious software. Third, it jeopardizes the entire ICS through the use of zero-day exposure on the OLE for Process Control (OPC) component. Fourth, it chooses to hide from ICS in order to control or inflict real damage on the physical infrastructure. Hax has been found to be effective in avoiding IDS and firewalls that are widespread in the energy, aerospace, pharmaceutical and petrochemical industries of the United States and Europe, resulting in economic losses of more than millions of dollars worldwide. In addition, many other well-known ICS entry events such as Stuxnet, Flame, Duqu, Black Energy etc. it is also organized into many categories and usually lasts a long time, leading to advanced threats (APT) in the targeted industries as well. in the end it has had a profound effect on our human society.

By comparing both models in finding multiple stage attacks is complex and very challenging.

In summary, we make three major contributions to this paper as follows:

1. We face the challenge of long-term dependence on multi-stage attack and suggest a sequence-to-sequence model (seq2seq) for this problem.

2. so we design our seq2seq model with encoder-decoder structure and use a short-term memory network (LSM) to build both the connector and the output to learn the long-term dependence.

3. we performed special tests on two public databases and compared those values with the HMM model to provide the best accuracy while using LSTM.

II. LITERATURE SURVEY

[1] This paper is the first situation that points to the weakness of current security systems by using virtualization to bypass these protections. The second scenario is about the level of attacks that are not addressed with the current defense solutions. This has moved us to look at the attack differently, where the source of the attack is of primary interest. In the third case, we focus on the infected computer in the process of registering other computers on the botnet. In each attack, they make an in-depth analysis of each step of the attack. [2] Early detection of multi-stage attacks is an important step in combating malware and shutting down the system. Many traditional security solutions use signature-based detection, which often fails to prevent zero-day attacks. Manual analysis of these samples requires a great deal of effort to successfully withstand the growth of unprofessional computer samples. In this paper, they present a novel mechanization and MITER Adversary Tactic Technique and a General Information Framework for the realization of multi-stage attacks in real time. First, they developed an operating system that receives notification while malicious resources are downloaded through a browser or a new process is introduced in the system. After the information is released, the engine releases the standalone features to be read if the utility is malicious. Second, they use the MITER ATT & CK framework, developed based on real-world cyberattack attacks, which best describes multi-stage attacks on enemy tactics, Tricks and Procedures to detect malicious activity and predict the stages a malicious computer program does during an attack. Finally, they propose a real-time system that combines both of these multi-stage attack detection methods. [3] In this paper, they present the method of detecting, visualizing, and predicting attackers' behavior patterns on a network-based system. have proposed a system that is able to detect a temporary intrusion pattern that reflects the behavior of attackers using warnings generated by the Intrusion Detection System (IDS). they use data mining techniques to detect warning patterns generated by creating Society rules. Their system is capable of broadcasting Snort real-time notifications and predicts entry based on our learned rules. Therefore, they are able to automatically detect patterns in multi-stage attacks, visualize patterns, and predict interference. [4] In this paper they discuss a method based on a sequential pattern mining system in order to find multi-phase invasive activity patterns to reduce work to create pattern rules. But in a flexible network environment where novel attack strategies are progressively emerging, a new approach they are proposing to use a rising mining algorithm demonstrates better capacity to detect emerging attacks. In order to improve the accuracy of the results and to shorten the working time of the mining algorithms, a targeted graph was introduced to limit the amount of data generated in the mining phase, which is particularly useful for rising mines. Finally, they remove the unintended consequences from the mines by computer-generated possible points between successive steps in a multi-stage attack pattern. A series of tests show the validity of the methods in this paper. [5] In this paper deals deals have found that the file contains a malicious link where the exploit and / or exploitation is downloaded from the hosting machine. This can be used to launch a targeted attack based on the victim's profile. We found out that the attacker was infecting the victims' machine with adware. The contribution of this paper is to detect novel attacks that a criminal can hide in innocent video files for the purpose of organizing targeted attacks in multiple stages.

III. PROPOSED TECHNOLOGY

In this section, the design of our seq2seq detection model, that include the modeling goals, basic assumptions, and a high level overview of our model .

2.1 Modeling goals :

Since our model is expected to be able to detect the sequence of the attack stage, to find the long-term dependence of the sections, we have two design objectives such as

1. Sequence detection: the main purpose of multi-stage attack detection is to detect attack sequences from the target.

2. Long-term dependence acquisition: as modern multi-stage acquisition solutions cannot effectively address the long-term dependency challenge, our main goal in this project is to overcome this challenge. As a result, our detection model should be built to be able to process long sequences of inputs and detect potential attack phases buried in such a long sequence.

2.2 Basic assumptions:
Since our seq2seq model operates in a series of IDS alerts as inputs, we create two ideas to address these needs:

1. We start by taking the basic IDS in front of our acquisition model. These conflicts are often stored in real-world online environments, as attackers who make multiple intruders are likely to target sensitive infrastructure that is already under tight security. In fact, when cybercriminals attempt targeted intervention without any protection (IDS), a single act of attack (e.g., exploitation without recognizing a known danger) usually works.

2. We also assume that the basic IDS is strong enough to capture sufficient warnings to represent any force.attack stages. With this in mind, we can focus on building sequential-to-sequence detection without worrying about input sequences may not detect attack stages. We acknowledge that this assumption may be particularly strong if the attackers use the risk of zero day in other stages of attack, but the discovery of zero-day attacks is outside of this paper. We are leaving work to find multi-stage attacks on zero-day exploitation in our upcoming research.

2.3 Model overview :
In this section, we present a comprehensive overview of how our seq2seq acquisition model works. For a clearer understanding, we use Fig. 1 as a directed example to illustrate. As can be seen from this figure, there is a significant infrastructure that has used IDS easily on its peripheral server which can continuously scan network packets from the internet and increase alerts on suspected detected packets. If attackers attempt to disrupt the infrastructure with multiple stages of attack, the IDS may capture individual attack actions and raise warnings accordingly. Our seq2seq detection model uses IDS alerts and uses the LSTM neural network (i.e., encoder) to encrypt the sequence of IDS alerted images into a hidden vector. After that, our seq2seq has another LSTM (i.e., decoder) to take this hidden vector as the original memory and record it for possible stage sequence. In this way, our seq2seq model can work with IDS

and enable the detection of multi-stage attacks without the need for further human analysis.

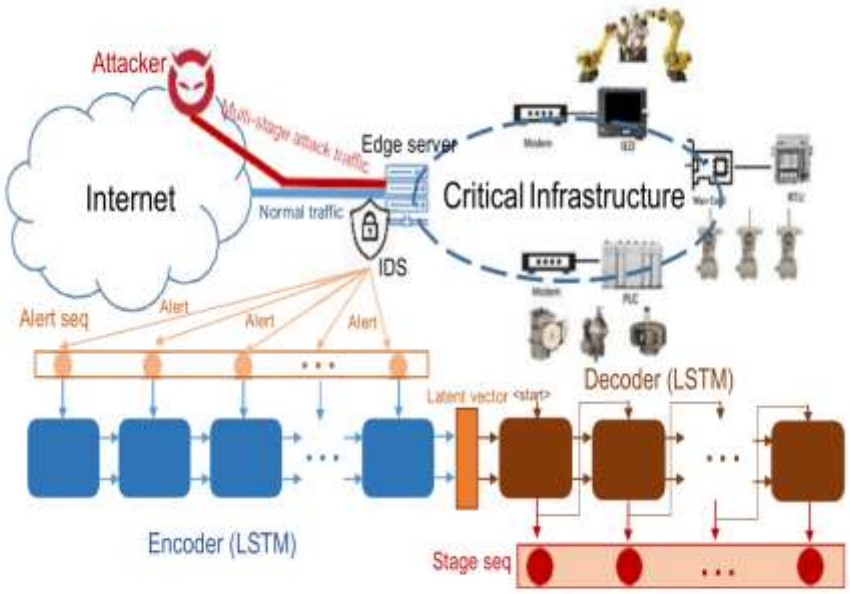

Fig. 1 – The overview of our seq2seq model for multi-stage attack detection.

To implement this project we have designed following modules

1) Upload Darpa/CSE-CIC Dataset: using this module we will upload dataset to application and this dataset available inside 'Datasets' folder
2) Preprocess Dataset: dataset contains both numeric and non-numeric datasets but algorithms accepts only numeric data so by apply ENCODER class we can convert non-numeric data to integer ID and then replace missing values with 0 and then split datasets into train and test where application used 80% dataset for training and 20% for testing
3) Run HMM Algorithm: using this module we will trained HMM algorithm by using 80% dataset and then trained a model and then this model will be applied on 20% test data to calculate TRUE POSITIVE PREDICTION (TPR) or correct prediction
4) Run LSTM SEQ2SEQ Algorithm: using this module we will trained LSTM encoder and decoder model on above dataset and then calculate its TPR value

Comparison Graph: using this module we will plot TPR graph between HMM and LSTM

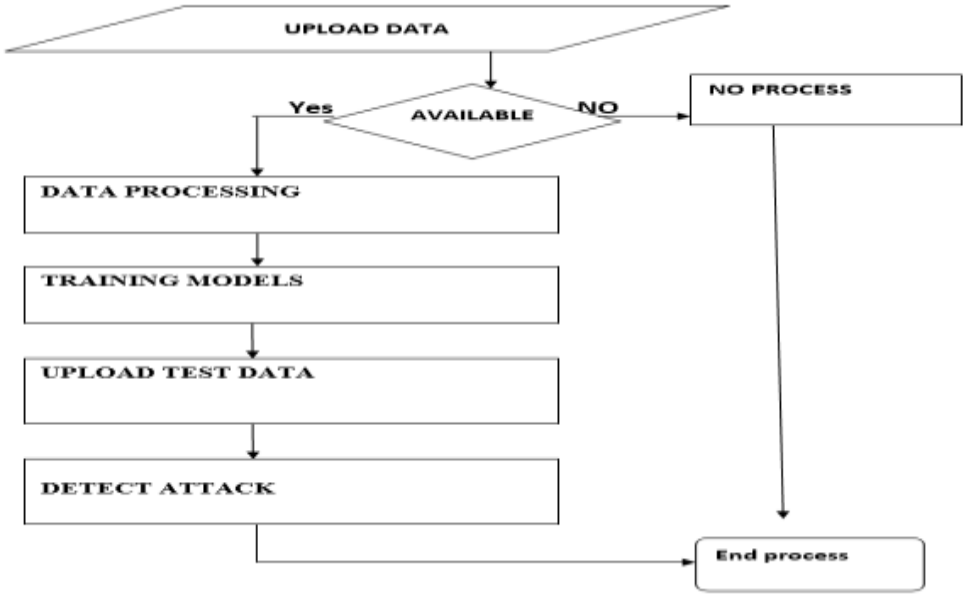

Fig .2 - Data flow diagram

IV. IMPLIMENTATION

We use our seq2seq acquisition model using Ten-sorFlow 2.3.0 using Keras 2.4.0 APIs instead of Python (version 3.6.9). For both the installer and the decoder, we call the Keras API keras.layers.Input () with shape = (None, Dim) to build the input layer at the beginning, where Dim = T + 1 (i.e., the number of warning types and the "end" symbol of the encoder and Dim = M + 3 (i.e., the number of potentially invasive categories and the “beginning”, “end” and “unknown” symbols in the output. We then use the function keras.layers.LSTM () to configure LSTM content in units = 64 in both the encoder and the decoder. Continuously, we stop return_state = True in LSTM encoder to get C (e) N and h (e) N, and bring it to the encoder by launching the LSTM encoder with initial_state = [C (e) N, h (e) N], using the connection from the encoder to the output. In the LSTM decoder, we call the function of Keras keras.layers.Dense () with activation = softmax to enable possible output for category detection. In order to use the multi-encoder solution, we simply duplicate the LSTM K times of the encoder and make a vector measurement of recovery over all encoders.

For comparison purposes, we also use the HMM detection model using the hmmlearn library version in 0.2.3 in Python. Specifically, we built the HMM model by calling it hmm = hmmlearn.MultinomialHMM () and implemented the parameters of the model hmm.startprob_ (initial opportunities), hmm.transmat_ (opportunities for change) and hmm.e-missionprob_ (output opportunities) by calculating directly from on the data sets we used for our experiments. Next, we use the viterbi algorithm by calling the function hmm.predict () to find the attack categories.

V. RESULTS AND DISCUSSION

Below screen showing LSTM encoder and decoder code to train SEQ2SEQ attack prediction

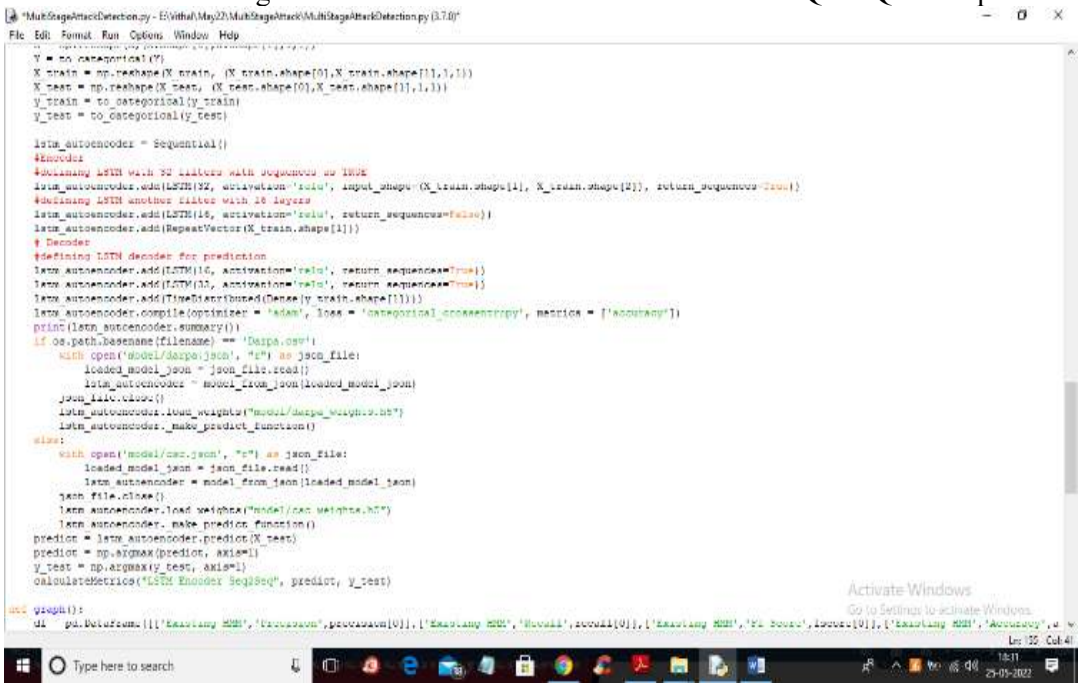

Fig .3 - read red colour comments to know about LSTM SEQ2SEQ training and prediction using encoder and decoder module

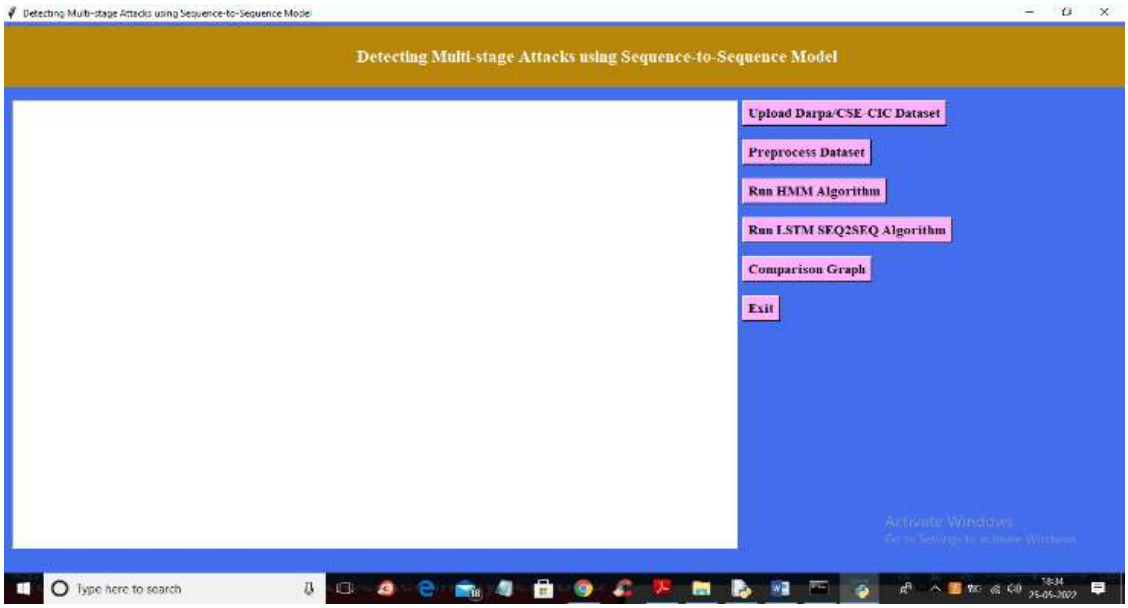

Fig . 4 – Homes screen

In above screen click on 'Upload Darpa/CSE-CIC Dataset' button to upload dataset and to get below screen

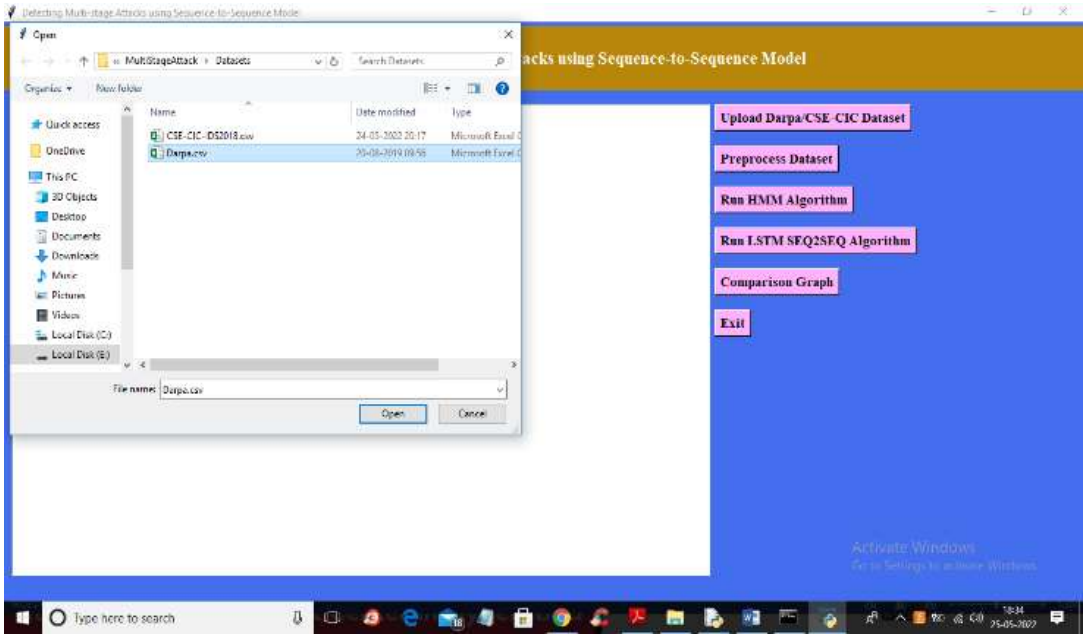

Fig .4 uploading data set

In above screen selecting and uploading 'Darpa.csv' file and then click on 'Open' button to load dataset and get below output

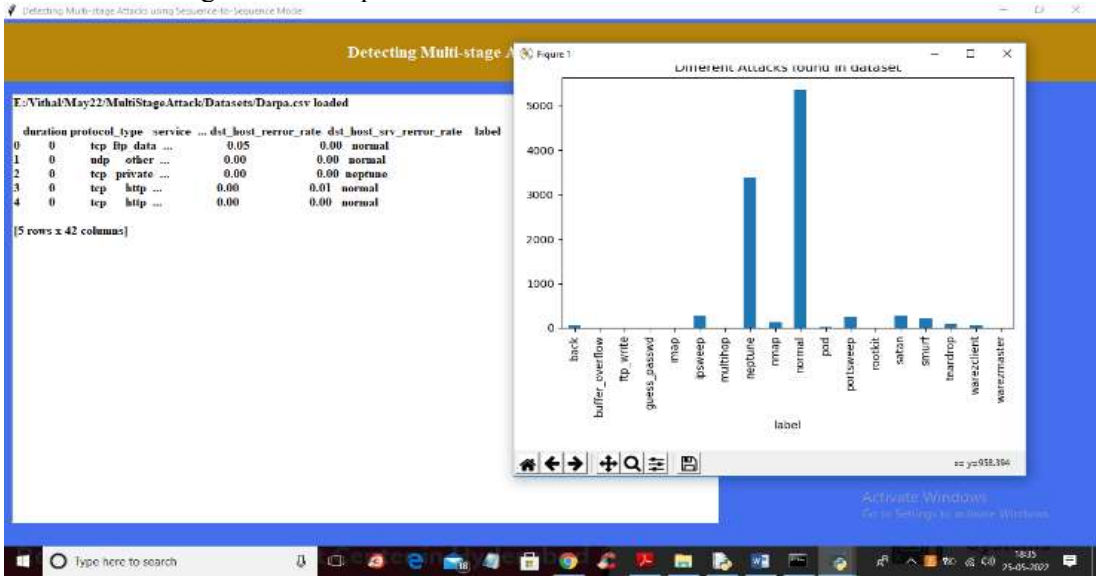

Fig .5 – Detected attacks in dataset

In above screen dataset loaded and we can see dataset contains both numeric and non-numeric values so we need to encode non-numeric data into numeric and in graph we can see x-axis contains different attacks and y-axis contains count of those attacks found in dataset. Now click on 'Preprocess Dataset' button to process dataset which will convert non-numeric data to numeric and get below output

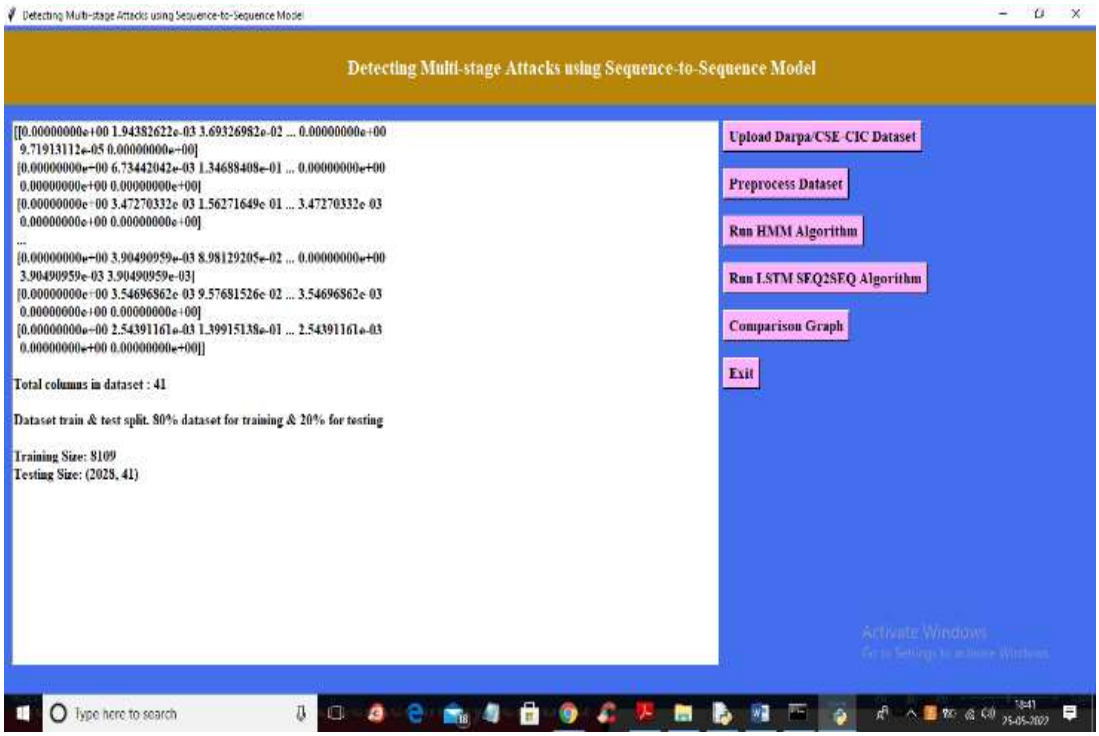

Fig .6 -preprocessing of dataset

In above screen we can see dataset converted to numeric format and application using 80% (8109) records for training and 2028
(20%) records for testing and dataset contains total 41 features or columns. Now train and test data is ready and now click on 'Run HMM Algorithm' button to train HMM and get below output

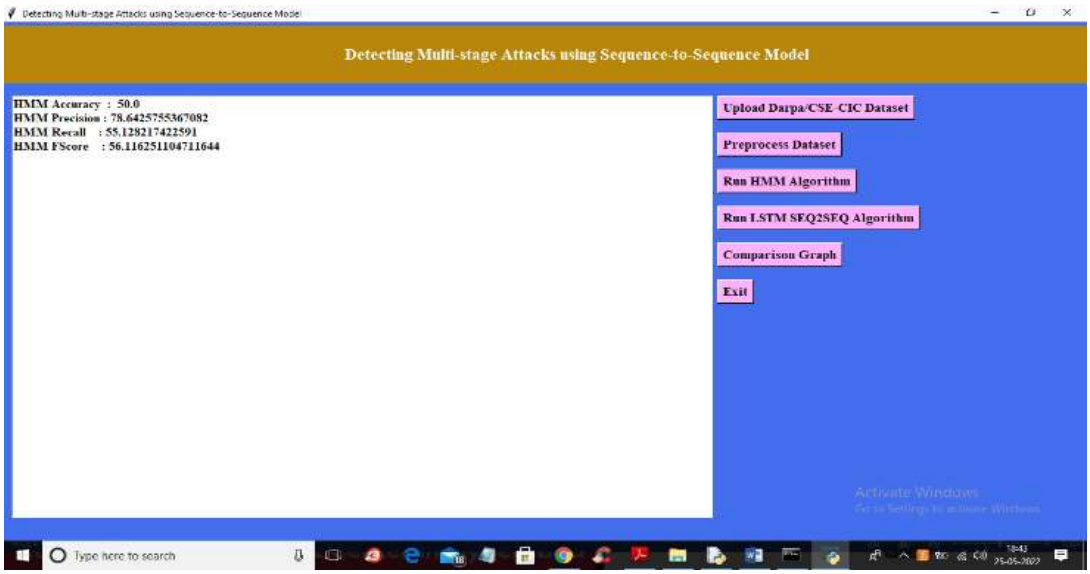

Fig .7 values which are obtained from HMM

In above screen with HMM we got correct attack prediction accuracy as 50% so HMM is not suitable to predict multi-stage attack as its accuracy is less and now click on 'Run LSTM SEQ2SEQ Algorithm' button to train LSTM and get below output

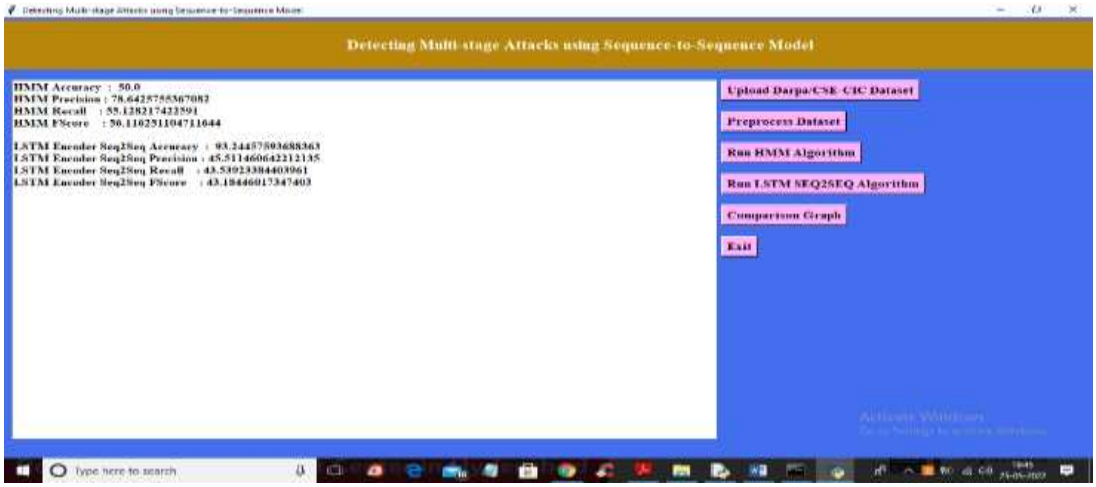

Fig .8 values which are obtained from LSTM

In above screen with LSTM we got correct attack prediction accuracy as 93% so LSTM is accurate in multi stage attack prediction and now click on 'Comparison Graph' button to get below graph

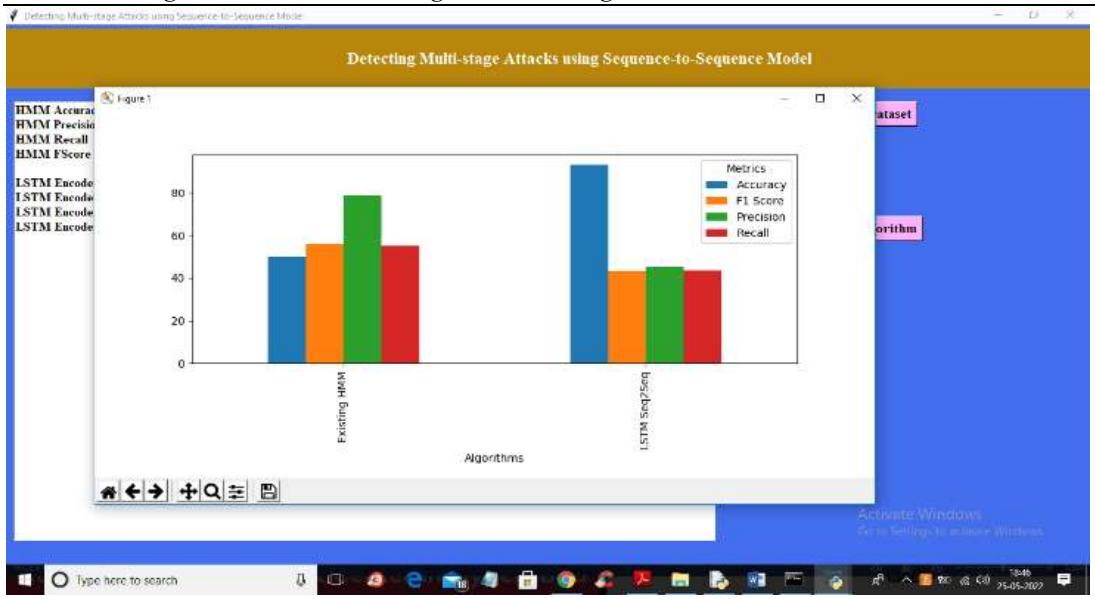

Fig .9 comparing both values of HMM and LSTM

In above screen x-axis contains algorithm names and y-axis contains accuracy and other metric values where each different colour bar represents different metric such as accuracy, precision, recall and FSCORE and in both algorithms LSTM accuracy is high.

VI. CONCLUSSION

In this paper, we are worked with sequence-to-sequence model for detecting multi-stage attacks. In Our model we applied LSTM to solve the detection challenge of long stage dependencies buried in the sequence of alerts, and also worked an encoder decoder architecture in order to take the full advantage of available knowledge that can be learned from the input alert sequences for detection. Finally we have compared our model with the state-of-the-art HMM solution, and confirmed a better detection performance, especially in the condition of the attacks with a longer stage dependencies.

VII. REFERENCES

[1] Pagna Diss, Jules. (2013). Toward a multistage attack detection framework.

[2] Katipally, Rajeshwar & Gasior, Wade & Cui, Xiaohui & Yang, Li. (2010). Multistage attack detection system for network administrators using data mining. 51. 10.1145/1852666.1852722.

[3] Takey, Yuvraj & Tatikayala, Sai & Sarma, Samavedam & Eswari, P & Patil, Mahesh. (2021). Real Time early Multi Stage Attack Detection. 283-290. 10.1109/ICACCS51430.2021.9441956.

[4] Li, Zhitang & Zhang, Aifang & Li, Dong & Wang, Li. (2007). Discovering Novel Multistage Attack Strategies. 4632. 45-56. 10.1007/978-3-540-73871-8_6

[5] Li, Zhitang & Zhang, Aifang & Li, Dong & Wang, Li. (2007). Discovering Novel Multistage Attack Strategies. 4632. 45-56. 10.1007/978-3-540-73871-8_6

[6] V Nath, Hiran & Mehtre, Babu. (2014). Video Files and Multistage Attacks:(Im)Possible???. 10.1109/INDICON.2014.7030520.

[7] Mathew, Sunu & Giomundo, Rich & Upadhyaya, Shambhu & Sudit, Moises & Stotz, Adam. (2006). Understanding multistage attacks by attack-track based visualization of heterogeneous event streams. 1-6. 10.1145/1179576.1179578.
[8] Berezinski, Przemyslaw & Piotrowski, Rafał & Śliwa, Joanna & Jasiul, Bartosz. (2012). Detection of Multistage Attack in Federation of System Environment.
[9] Zhang, Ai-fang & Li, Zhi-tang & Li, Dong & Wang, Li. (2007). Discovering Novel MultistageAttack Patterns in Alert Streams. 115-121. 10.1109/NAS.2007.20.
[10] Shao, Chengwu & Li, Yan-Fu. (2021). Multistage Attack-Defense Graph Game Analysis for Protection Resources Allocation Optimization Against Cyber Attacks Considering Rationality Evolution. Risk analysis : an official publication of the Society for Risk Analysis. 10.1111/risa.13837.